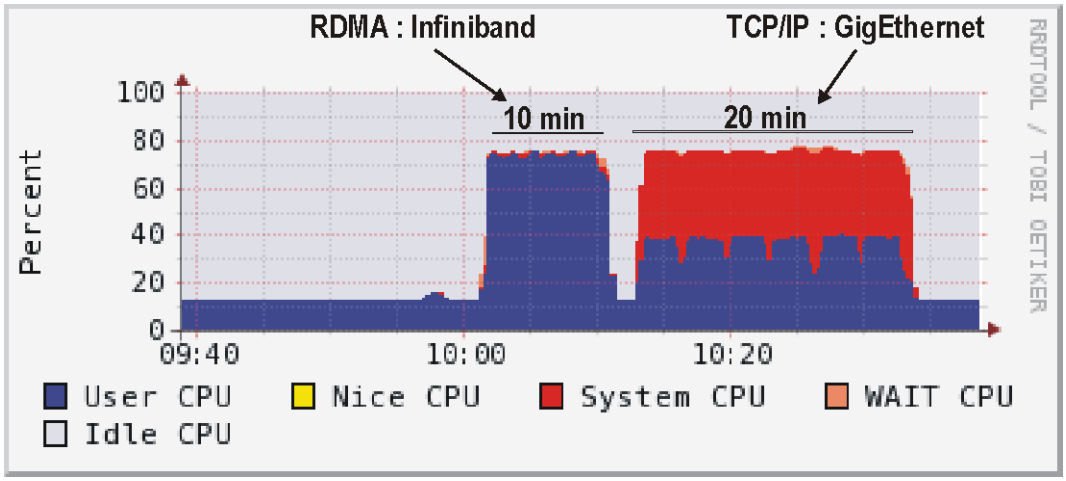
NOTE TECHNIQUE DSI

Institut de recherche pour le développement

CLUSTER DE CALCUL DU CENTRE DE NOUMEA

UPGRADE INFINIBAND : RETOUR D'EXPERIENCE



Auteurs

Jérôme Lefèvre, resp. Informatique UMR 65 / UR 103 (jlefevre@noumea.ird.nc)

Michel Ménézo, resp. SIL Nouméa (menezo@noumea.ird.nc)

Avril 2009

Version 1.0

SOMMAIRE

1. Objet et contexte 4
2. Limites du réseau Gigabit/Ethernet 5
3. Matériel InfiniBand installé sur le cluster 7
4. Installation du support logiciel pour InfiniBand OFED 8
5. Installation de la librairie openMPI avec support InfiniBand 16
6. Post-Installation : autres paramètres système à ajuster 19
7. Gains apportés par l'InfiniBand 20
8. Pistes d'optimisation de l'architecture de calcul de Nouméa 26

ANNEXES

ANNEXE A Présentation des activités de modélisation en grande partie réalisées sur le cluster de Nouméa et exemple de portage de l'expérience vers le Pérou

ANNEXE B Présentation abrégée du switch InfiniBand Flextronics et des composants IBA : carte InfiniHost et câbles

ANNEXE C Installation du cluster ToolKit OSCAR : événements postés sur les forums OSCAR

ANNEXE D Présentation abrégée du paquetage Mellanox OpenFabrics Enterprise Distribution for Linux (MLNX_OFED)

ANNEXE E Fichier de configuration "ofed.conf" définissant les modules et drivers IBA à installer dans le processus d'installation MLNX_OFED

Couverture : Illustration de l'Overhead des 20 processeurs pour la configuration ROMS (Safe) avec à gauche, l'échange des messages par réseau InfiniBand et à droite l'échange par réseau GigEthernet / TCP

Liste des tableaux

Tableau 1 : Performances des principaux réseaux de communication employés dans les fermes de PC 5

Tableau 2 Description et prix du matériel InfiniBand commandé chez XENON 7

Liste des figures

Figure 1 Cluster de PC installé à Nouméa, architecture Opteron 285 et réseau GigE + InfiniBand. 40 CPUs 4

Figure 2 Evolution du prix de l'équipement en réseau rapide (InfiniBand et/ou Myrinet) pour 10 noeuds (switch + adaptateurs). (Informations obtenues depuis les forums www.beowulf.org et l'intégrateur Alinéos) 5

Figure 3 Illustration de l'overhead des processeurs d'un nœud de calcul sur une configuration ROMS employant successivement 1, 2, 3 et 4 nœuds (4 CPUs/nœud). 6

Figure 4 Emprise géographique de la configuration ROMS (Safe), représentée ici par le champs de température en surface 20

Figure 5 Emprise géographique de la configuration régionale WRF, représentée ici par la topographie (mètres). Le rectangle intérieur symbolise l'emprise du second domaine emboité. 20

Figure 6 Emprise géographique de la configuration régionale MARS, représentée ici par la topographie (mètres) 21

Figure 7 Illustration de l'Overhead des 20 processeurs pour la configuration ROMS (Safe) avec à gauche, l'échange par réseau InfiniBand et à droite l'échange par réseau GigE 22

Figure 8 Comparaison des temps de calcul pour la configuration ROMS Safe selon différents modes de partitionnement et modes de transfert entre l'architecture de Cape Town et celle de Nouméa. 22

Figure 9 Courbes de Speed-Up du code ROMS (Safe) sur l'architecture IBA Cape Town (rouge) et l'architecture IBA et GigE de Nouméa 23

Figure 10 Comparaison des temps de calcul pour la configuration WRF selon différents modes de partitionnement et pour les deux modes d'échanges GigE TCP et InfiniBand 24

Figure 11 Courbes de Speed-Up du code MARS sur l'architecture IBA et GigE de Nouméa 24

1. Objet et contexte

Cette note technique s'attache à présenter les gains obtenus depuis l'upgrade du réseau de communication du cluster de Nouméa, ainsi que le détail du matériel InfiniBand nouvellement installé sur le cluster et la procédure simplifiée d'installation.

Pour mémoire, le projet d'upgrade du réseau de communication a fait l'objet d'une précédente demande au titre de l'opération SPIRALES soutenue par la DSI, détaillée dans le document ''Projet Spirales 2008'' :

> ***Projet Spirales 2008** : Upgrade du Cluster de Nouméa, Réseau rapide de communication, Novembre 2008.*

Mis en production en janvier 2005, le cluster de Nouméa constitue une plateforme d'investigations numériques adaptée aux activités de modélisation océanique et atmosphérique régionale conduites localement par les 3 équipes de Recherche : UMR LEGOS (IRD-UR65), CAMELIA (IRD-UR103) et LOCEAN (IRD-UR182). L'annexe A présente un bilan des activités de modélisation réalisées en grande partie au moyen de la plateforme de calcul de Nouméa. Les principales investigations numériques conduites sur le cluster concernent :

- *Etude des processus et répercussion de la variabilité climatique dans le Pacifique Sud-Ouest*
- *Relation thonidès et Environnement dans le Pacifique Sud-Ouest*
- *Risques et Aléas Cycloniques en relation avec le climat*
- *Modélisation côtière et lagonaire haute résolution*
- *Océanographie opérationnelle dans le cadre du chantier Prévimer*
- *Couplage régional Océan/Atmosphère*

Dans le cadre de ces investigations, les codes WRF, ROMS et ROMS-PISCES, MARS, ADCIRC et FVCOM ont été portés sur le cluster dans leur version parallélisée. Courant 2009, le code OPA devrait également être porté sur le cluster.

Le cluster de Nouméa comporte 20 CPUs Opteron 285 bicoeur, procurant au total 40 CPUs et une capacité mémoire vive voisine de 60 Gb. Le réseau de communication Ethernet/Gigabit a été identifié comme le principal goulet d'étranglement, ce qui a motivé l'upgrade en réseau InfiniBand en 2008.

Figure 1 Cluster de PC installé à Nouméa, architecture Opteron 285 et réseau GigE + InfiniBand. 40 CPUs

2. Limites du réseau Gigabit/Ethernet

L'architecture de calcul installée au centre de Nouméa, sous la supervision de Michel Ménézo et Jérôme Lefèvre, se compose de 10 serveurs Tyan S2885 bi-processeurs à base d'Opteron 285 double cœur. Jusqu'à récemment, le réseau de communication inter-noeuds reposait sur la technologie Gigabit Ethernet (GigE) et employait le protocole TCP/IP pour l'échange des messages. Cette technologie de communication inter-nœud est la plus répandue car la moins onéreuse, mais en contrepartie la moins performante des réseaux de communication. Comme le montre le Tableau 1, les performances du GigE sont très en deçà de celles procurées par les réseaux rapides InfiniBand ou Myrinet généralement employés dans les fermes de calcul HPC.

Performance	InfiniBand	Myrinet	Gigabit Ethernet
Switch Latency	160 ns	200 ns	10,000 ns
End-to-End Latency (8 byte packet)	7.6 us*	8 us	60 us
End-to-End Latency (1K byte packet)	13 us	22 us	80 us
Throughput	822 MB/sec	250 MB/sec	100 MB/sec
CPU Overhead	3%	6%	80%

* Planned software/firmware releases will further reduce latency. Data provided by: Prof Dhabaleswar K. Panda, Ohio State University, Jan. 2003

Tableau 1 : Performances des principaux réseaux de communication employés dans les fermes de PC

La technologie GigE a été initialement retenue pour le réseau, afin de ne pas grever le coût initial de l'équipement. Depuis la mise en service de notre cluster début 2005, le coût de l'équipement en technologie InfiniBand a fortement chuté (Cf Figure 2), ce qui rend cet investissement aujourd'hui pertinent, en plus de la réponse qu'il apporte pour les configurations à base de stations multi-cœurs. Le réseau GigE devient rapidement le principal goulet d'étranglement pour les applications parallélisées intensives en échange de données et employant des messages de grandes tailles (Cf Projet Spirales 2008, Annexe C)

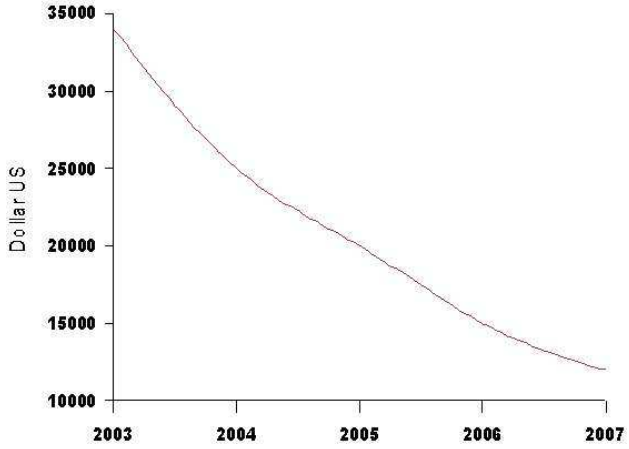


Figure 2 Evolution du prix de l'équipement en réseau rapide (InfiniBand et/ou Myrinet) pour 10 noeuds (switch + adaptateurs). (Informations obtenues depuis les forums www.beowulf.org et l'intégrateur Alinéos)

D'autre part, la communication reposant sur le protocole TCP/IP, celui-ci induit des consommations CPU lourdes et des pertes de temps en raison des copies systématiques dans une mémoire tampon : le socket buffer. Les interruptions fréquentes du processeur, la perte de cycles CPU (Overhead[1]) réduisent la performance globale du système de calcul.

Nous rencontrons actuellement ce problème sur notre cluster pour nos grosses configurations ROMS-PISCES, WRF et également avec le modèle MARS, dont la stratégie de parallélisation employée dans ce dernier implique l'échange de paquets de grande taille, ce qui est très pénalisant avec le réseau GigE, dont la performance reste satisfaisante seulement dans la gamme des paquets de petites taille (Cf *Projet Spirales 2008 Annexe C)*. La Figure 3 illustre l'overhead des CPUs à mesure que le partitionnement de la grille de calcul augmente, impliquant une augmentation de l'échange des données. Pour une configuration mettant à contribution 2 nœuds de calcul, à raison de 4 processeurs par nœud, les processeurs affichent chacun 30 % de perte de cycle CPU à cause des interruptions liées à la gestion des communications. A partir de 3 nœuds, la perte atteint 60 %, ce qui engendre une dégradation de la scalabilité[2] du code ROMS sur notre cluster équipé en réseau GigE. Le code MARS montre le même comportement, mais la limite de scalabilité du code est plus rapidement atteinte à cause de la taille des messages plus grande.

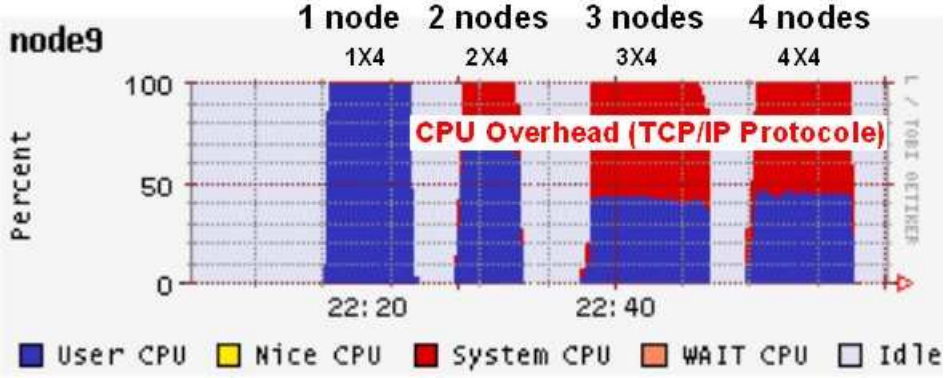


Figure 3 Illustration de l'overhead des processeurs d'un nœud de calcul sur une configuration ROMS employant successivement 1, 2, 3 et 4 nœuds (4 CPUs/nœud).

Avec l'arrivée des stations multi-coeurs, la demande en entrée/sortie est très significativement augmentée, impliquant une sollicitation accrue du processeur liée à la gestion des échanges de données. La solution est de passer par une architecture réseau, n'employant pas le protocole TCP/IP et ne sollicitant pas le processeur, ce dernier restant entièrement dédié au calcul. C'est ce que procurent comme avantage les réseaux basés sur la technologie InfiniBand ou Myrinet.

[1] Overhead est le temps passé (perdu) à effectuer une commutation entre deux processus en temps partagé. C'est le temps nécessaire au commutateur pour sauvegarder un contexte et en recharger un autre.

[2] Scalabilité est la capacité d'un système à évoluer en cas de montée en charge

3. Matériel InfiniBand installé sur le cluster

Les cartes mères Tyan S2885 des serveurs du cluster intègrent le chipset AMD-8131, lequel offre une possibilité d'interconnexion haut débit et rapide au moyen de 2 connecteurs PCI-X 64 bits 133 Mhz (bande passante disponible par slot : 6,4 GBits/s). Ces slots autorisent l'usage d'adaptateurs pour réseaux rapides, tels Myrinet, InfiniBand ou Ethernet 10 Gb.

Notre choix de technologie a porté sur l'InfiniBand, plus répandue, moins onéreuse et offrant des performances comparables à la technologie Myrinet. De plus, elle est supportée dans les différentes librairies de communication MPI que nous employons actuellement dans nos applications parallélisées.

Après une consultation locale et internationale, le matériel InfiniBand détaillé dans le Tableau 2 a été commandé en juillet 2008 auprès de la société XENON Systems, basée en Australie. Cette société est revendeur du matériel Mellanox, l'un des principaux leaders en technologie InfiniBand.

10 Adaptateurs HCA PCI-X Mellanox InfiniHost 10 GB/s 4X (MT23108 Cougar Cub 128 MB)	4 500 Euros
10 cables InfiniBand 4X	880 Euros
1 Switch InfiniScale 2 Gb DDR 4X 24 ports Flextronics (F-X430046)	3 740 Euros
Total	**9120 Euros**

Tableau 2 Description et prix du matériel InfiniBand commandé chez XENON

Le coût de cette opération 9120 Euros (hors taxe) est à rapprocher avec la proposition initiale de la société ALINEOS France de 8790 Euros (hors taxe) établie le 05/11/2007 et présentée dans le ''projet Spirales 2008, Annexe A''. Le matériel initialement pressenti n'a pu être acheté chez ALINEOS en raison de la rupture des adaptateurs PCI-X sur le marché Européen.

L'annexe B présente une description abrégée du switch Flextronics. La documentation du switch Flextronics est également disponible sur le site http://ibswitches.com.

4. Installation du support logiciel pour InfiniBand OFED

Environnement d'installation

L'accueil début 2008 d'Augusto Ingunza (IMARPE) dans le cadre du portage de l'expérience cluster au Pérou (Voir Annexe A) et la perspective de l'upgrade vers l'InfiniBand ont motivé la réinstallation complète du cluster de Nouméa.

L'installation du support logiciel InfiniBand a été réalisée sur une base Fedora Core 8 64bits augmentée de la couche logiciel Cluster ToolKit Oscar 5.1beta. Les informations utiles comportant à la fois le manuel d'installation, le guide d'administration et l'aide au démarrage sont disponibles sur le site http://svn.oscar.openclustergroup.org/trac/oscar

De plus, les forums Oscar-devel et Oscar-users sont très réactifs :

- http://www.mail-archive.com/oscar-devel@lists.sourceforge.net/
- http://www.mail-archive.com/oscar-users@lists.sourceforge.net/

Pour illustration, depuis la première installation du Cluster ToolKit Oscar 4.1 en 2005 et jusqu'à la récente réinstallation FC8 + Oscar 5.1beta, plusieurs posts ont été déposés sur les forums OSCAR. La liste présentée en Annexe C retrace ces événements et les retours de la communauté OSCAR.

Le cluster comprend les 10 serveurs Opteron 285, équipés chacun d'une carte Intel Pro1000 (connecteur GigE) et d'une carte HCA PCI-X InfiniHost. Le cluster comprend un réseau de communication GigE via un switch 3COM, lequel est utilisé pour le transfert des ordres d'administration et le partage des fichiers par NFS-UDP à travers le cluster et un réseau InfiniBand, qui après installation sera utilisé pour les échanges des messages associés aux calculs parallélisés.

Les composants principaux de l'InfiniBand Architecture (IBA)

IBA est un standard d'industrie qui définit un nouveau sous-système grande vitesse à switches conçu pour connecter les noeuds de processeurs et les noeuds E/S pour former un SAN (System Area Network). Cette nouvelle méthode d'interconnexion passe d'un modèle d'E/S basé sur des transactions locales via des bus à un modèle d'envoi de messages à distance via des canaux. Outre les éléments matériels composant l'IBA (carte hôte IBA, Switch IBA etc…), IBA comprend des éléments logiciels et des pilotes. Ces éléments sont rassemblés dans le paquetage OFED ''Mellanox OpenFabrics Enterprise Distribution for Linux'' (MLNX_OFED), téléchargeable sur http://www.mellanox.com. En outre, les sites http://www.openib.org et https://wiki.openfabrics.org offrent d'autres liens vers la documentation utile et des tutoriels portant sur la construction des paquetages OFED, etc.

La documentation introduisant MLNX_OFED est présenté en Annexe D. La documentation plus complète (guide d'installation et manuel d'utilisation) est disponible en suivant ces liens :

- http://www.mellanox.com/related-docs/prod_software/Mellanox_OFED_Installation_Guide.pdf
- http://www.mellanox.com/related-docs/prod_software/Mellanox_OFED_Linux_user_manual_1_40_1.pdf

Le paquetage MLNX_OFED contient le support du pilote de bas niveau pour les contrôleurs d'hôtes basés sur Mellanox (contrôleurs HCA), les protocoles niveau supérieur SDP

(Sockets Direct Protocol), le pilote de protocole de mise en réseau TCP/IP IPoIB (IP over InfiniBand) et le pilote de protocole SCSI Remote Direct Memory Access (RDMA). En outre, MLNX_OFED comprend également les utilitaires d'administration : OpenSM (subnet manager), IBadm (administration IB), des utilitaires diagnostiques/performances et des librairies MPI et MVAPICH spécialement compilées pour le support IB. Ces dernières librairies ne seront pas utilisées, la librairie OpenMPI préférée sur le cluster de Nouméa sera spécialement installée en activant le support InfiniBand.

Les étapes d'installation du support IBA au moyen de MLNX_OFED

- **Etape 1 : récupération du paquetage MLNX_OFED**

Sur la page Download, rubrique MLNX_OFED (accessible en tapant ce mot clé) du site http://www.mellanox.com, plusieurs isos MLNX_OFED_LINUX sont disponibles pour les plateformes Redhat. En téléchargeant l'un d'entre eux, on accède aux sources, contenues dans le répertoire /src. Les binaires inclus dans l'iso n'étant pas compatibles avec notre version OS et architecture x86_64, les sources permettront de reconstruire les RPMs spécifiquement pour notre architecture.

- **Etape 2 : Construction des RPMs depuis les sources et installation sur le nœud maître**

Copie du répertoire src vers /home/cluster/ofed

On s'assure que /src contienne au moins :

```
$ ls /home/cluster/ofed/src

mft-X.X.X-rcXX.tgz : L'utilitaire pour le flash du firmware des HCA Mellanox
OFED-X.X-mlnxX.tgz : contient les softs provenant d'Openfabric et les modules :
        o  OpenFabrics core and ULPs
             - HCA drivers (mthca, mlx4, ipath, ehca)
             - iWARP driver (cxgb3, nes)
             - core
             - Upper Layer Protocols: IPoIB, SDP, SRP Initiator, iSER Initiator
             RDS, qlgc_vnic and uDAPL
        o  OpenFabrics utilities
             - OpenSM: InfiniBand Subnet Manager
             - Diagnostic tools
             - Performance tests
        o  Extra packages
             - open-iscsi: open-iscsi initiator with iSER support
             - ib-bonding: Bonding driver for IPoIB interface
        o  Sources of all software modules (under conditions mentioned in the
        modules' LICENSE files)
        o  Documentation
```

Préalable indispensable : Lire le **README.txt**. Toutes les étapes conduisant au déploiement et au démarrage du support IBA y sont décrites.

Il faut simplement exécuter le script de configuration et de reconstruction des rpms :

```
$ ./install.pl
```

Une liste d'argument moyennant le fichier ''ofed.conf'' (Cf Annexe E, script ofed_editr.conf)

peut être passée au script install.pl de manière à sélectionner les pilotes et modules IBA utiles et désélectionner le support des périphériques n'ont présents sur l'architecture.

A la fin du processus d'installation (et dans le meilleur des cas !), les rpms sont automatiquement reconstruits pour la plateforme puis installés. L'utilitaire s'occupe également de charger les modules (vérifier leur chargement via la commande ''lsmod'') :

```
ib_ipoib, ib_sa, ib_uverbs, ib_umad, ib_mthca, ib_mad, ib_core
```

Par contre, il faut rebooter le nœud maître pour que le driver des cartes HCA soit pris en compte.

A l'issue de cette étape, l'installation de OFED sur le noeud maître a permis de construire des RPMS compatibles avec l'architecture. Ces RPMS se trouvent sous :

```
$ ls /home/cluster/ofed/src/OFED-X.X-mlnx1/RPMS/fedora-release-8-3/x86_64/
```

dapl-1.2.5-1.ofed1.3.mlnx1.x86_64.rpm
dapl-2.0.7-1.ofed1.3.mlnx1.x86_64.rpm
dapl-debuginfo-1.2.5-1.ofed1.3.mlnx1.x86_64.rpm
dapl-debuginfo-2.0.7-1.ofed1.3.mlnx1.x86_64.rpm
dapl-devel-1.2.5-1.ofed1.3.mlnx1.x86_64.rpm
dapl-devel-2.0.7-1.ofed1.3.mlnx1.x86_64.rpm
dapl-devel-static-1.2.5-1.ofed1.3.mlnx1.x86_64.rpm
dapl-devel-static-2.0.7-1.ofed1.3.mlnx1.x86_64.rpm
dapl-utils-1.2.5-1.ofed1.3.mlnx1.x86_64.rpm
dapl-utils-2.0.7-1.ofed1.3.mlnx1.x86_64.rpm
ibsim-0.4-1.ofed1.3.mlnx1.x86_64.rpm
ibsim-debuginfo-0.4-1.ofed1.3.mlnx1.x86_64.rpm
ibutils-1.2-1.ofed1.3.mlnx1.x86_64.rpm
ibutils-debuginfo-1.2-1.ofed1.3.mlnx1.x86_64.rpm
infiniband-diags-1.3.6-1.ofed1.3.mlnx1.x86_64.rpm
infiniband-diags-debuginfo-1.3.6-1.ofed1.3.mlnx1.x86_64.rpm
iscsi-initiator-utils-2.0-865.15.x86_64.rpm
kernel-ib-1.3-2.6.23.1_42.fc8.x86_64.rpm
kernel-ib-1.3-2.6.23.14cores_freq_off.x86_64.rpm
kernel-ib-devel-1.3-2.6.23.1_42.fc8.x86_64.rpm
kernel-ib-devel-1.3-2.6.23.14cores_freq_off.x86_64.rpm
libcxgb3-1.1.4-1.ofed1.3.mlnx1.x86_64.rpm
libcxgb3-debuginfo-1.1.4-1.ofed1.3.mlnx1.x86_64.rpm
libcxgb3-devel-1.1.4-1.ofed1.3.mlnx1.x86_64.rpm
libibcm-1.0.2-1.ofed1.3.mlnx1.x86_64.rpm
libibcm-debuginfo-1.0.2-1.ofed1.3.mlnx1.x86_64.rpm
libibcm-devel-1.0.2-1.ofed1.3.mlnx1.x86_64.rpm
libibcommon-1.0.8-1.ofed1.3.mlnx1.x86_64.rpm
libibcommon-debuginfo-1.0.8-1.ofed1.3.mlnx1.x86_64.rpm
libibcommon-devel-1.0.8-1.ofed1.3.mlnx1.x86_64.rpm
libibcommon-static-1.0.8-1.ofed1.3.mlnx1.x86_64.rpm
libibmad-1.1.6-1.ofed1.3.mlnx1.x86_64.rpm
libibmad-debuginfo-1.1.6-1.ofed1.3.mlnx1.x86_64.rpm
libibmad-devel-1.1.6-1.ofed1.3.mlnx1.x86_64.rpm
libibmad-static-1.1.6-1.ofed1.3.mlnx1.x86_64.rpm
libibumad-1.1.7-1.ofed1.3.mlnx1.x86_64.rpm
libibumad-debuginfo-1.1.7-1.ofed1.3.mlnx1.x86_64.rpm
libibumad-devel-1.1.7-1.ofed1.3.mlnx1.x86_64.rpm
libibumad-static-1.1.7-1.ofed1.3.mlnx1.x86_64.rpm
libibverbs-1.1.1-1.ofed1.3.mlnx1.i686.rpm
libibverbs-1.1.1-1.ofed1.3.mlnx1.x86_64.rpm
libmthca-1.0.4-1.ofed1.3.mlnx1.x86_64.rpm
libmthca-debuginfo-1.0.4-1.ofed1.3.mlnx1.x86_64.rpm
libmthca-devel-static-1.0.4-1.ofed1.3.mlnx1.x86_64.rpm
libnes-0.5-1.ofed1.3.mlnx1.x86_64.rpm
libnes-debuginfo-0.5-1.ofed1.3.mlnx1.x86_64.rpm
libnes-devel-static-0.5-1.ofed1.3.mlnx1.x86_64.rpm
librdmacm-1.0.6-1.ofed1.3.mlnx1.x86_64.rpm
librdmacm-debuginfo-1.0.6-1.ofed1.3.mlnx1.x86_64.rpm
librdmacm-devel-1.0.6-1.ofed1.3.mlnx1.x86_64.rpm
librdmacm-utils-1.0.6-1.ofed1.3.mlnx1.x86_64.rpm
libsdp-1.1.99-1.ofed1.3.mlnx1.x86_64.rpm
libsdp-debuginfo-1.1.99-1.ofed1.3.mlnx1.x86_64.rpm
libsdp-devel-1.1.99-1.ofed1.3.mlnx1.x86_64.rpm
mpi-selector-1.0.0-1.x86_64.rpm
mpitests_mvapich2_gcc-3.0-773.x86_64.rpm
mpitests_mvapich2_intel-3.0-773.x86_64.rpm
mpitests_mvapich2_pgi-3.0-773.x86_64.rpm
mpitests_mvapich_gcc-3.0-773.x86_64.rpm
mpitests_mvapich_intel-3.0-773.x86_64.rpm
mpitests_mvapich_pgi-3.0-773.x86_64.rpm
mpitests_openmpi_gcc-3.0-773.x86_64.rpm
mpitests_openmpi_intel-3.0-773.x86_64.rpm
mpitests_openmpi_pgi-3.0-773.x86_64.rpm
mstflint-1.3-1.ofed1.3.mlnx1.x86_64.rpm
mstflint-debuginfo-1.3-1.ofed1.3.mlnx1.x86_64.rpm
mvapich2_gcc-1.0.2-4.x86_64.rpm
mvapich2_intel-1.0.2-4.x86_64.rpm
mvapich2_pgi-1.0.2-4.x86_64.rpm
mvapich_gcc-1.0.0-2106.x86_64.rpm
mvapich_intel-1.0.0-2106.x86_64.rpm
mvapich_pgi-1.0.0-2106.x86_64.rpm
ofed-docs-1.3-0.x86_64.rpm
ofed-scripts-1.3-0.x86_64.rpm
open-iscsi-2.0-865.15.x86_64.rpm
open-iscsi-generic-debuginfo-2.0-865.15.x86_64.rpm
openmpi_gcc-1.2.5-1.x86_64.rpm
openmpi_intel-1.2.5-1.x86_64.rpm
openmpi_pgi-1.2.5-1.x86_64.rpm
opensm-3.1.10-1.ofed1.3.mlnx1.x86_64.rpm
opensm-debuginfo-3.1.10-1.ofed1.3.mlnx1.x86_64.rpm
opensm-devel-3.1.10-1.ofed1.3.mlnx1.x86_64.rpm
opensm-libs-3.1.10-1.ofed1.3.mlnx1.x86_64.rpm

```
libibverbs-debuginfo-1.1.1-1.ofed1.3.mlnx1.i686.rpm
libibverbs-debuginfo-1.1.1-1.ofed1.3.mlnx1.x86_64.rpm
libibverbs-devel-1.1.1-1.ofed1.3.mlnx1.i686.rpm
libibverbs-devel-1.1.1-1.ofed1.3.mlnx1.x86_64.rpm
libibverbs-devel-static-1.1.1-1.ofed1.3.mlnx1.i686.rpm
libibverbs-devel-static-1.1.1-1.ofed1.3.mlnx1.x86_64.rpm
libibverbs-utils-1.1.1-1.ofed1.3.mlnx1.i686.rpm
libibverbs-utils-1.1.1-1.ofed1.3.mlnx1.x86_64.rpm
libipathverbs-1.1-1.ofed1.3.mlnx1.x86_64.rpm
libipathverbs-debuginfo-1.1-1.ofed1.3.mlnx1.x86_64.rpm
libipathverbs-devel-1.1-1.ofed1.3.mlnx1.x86_64.rpm
libmlx4-1.0-1.ofed1.3.mlnx1.x86_64.rpm
libmlx4-debuginfo-1.0-1.ofed1.3.mlnx1.x86_64.rpm
libmlx4-devel-static-1.0-1.ofed1.3.mlnx1.x86_64.rpm
opensm-static-3.1.10-1.ofed1.3.mlnx1.x86_64.rpm
perftest-1.2-1.ofed1.3.mlnx1.x86_64.rpm
perftest-debuginfo-1.2-1.ofed1.3.mlnx1.x86_64.rpm
qperf-0.4.0-1.ofed1.3.mlnx1.x86_64.rpm
qperf-debuginfo-0.4.0-1.ofed1.3.mlnx1.x86_64.rpm
rds-tools-1.1-2.x86_64.rpm
rds-tools-debuginfo-1.1-2.x86_64.rpm
sdpnetstat-1.60-1.ofed1.3.mlnx1.x86_64.rpm
sdpnetstat-debuginfo-1.60-1.ofed1.3.mlnx1.x86_64.rpm
srptools-0.0.4-1.ofed1.3.mlnx1.x86_64.rpm
srptools-debuginfo-0.0.4-1.ofed1.3.mlnx1.x86_64.rpm
tvflash-0.9.0-1.ofed1.3.mlnx1.x86_64.rpm
tvflash-debuginfo-0.9.0-1.ofed1.3.mlnx1.x86_64.rpm
```

Ces pilotes et utilitaires seront installés ultérieurement sur les nœuds de calcul, sauf le gestionnaire Open Subnet Manager (openSM). Au moins une machine sur tout le réseau InfiniBand doit exécuter un gestionnaire sous-réseau pour configurer le routage de connexion de matériel de bas niveau. Ce dernier est laissé en exécution comme démon système pour traiter la reconfiguration de la structure de sous-réseau (subnet fabric) dans le cas d'événements de changement du statut de la connexion. Par conséquent, openSM actif sur le noeud maître suffit pour l'administration.

- **Etape 3 : Démarrage des services et vérification de la prise en charge du matériel IBA sur le nœud maître**

Plusieurs utilitaires de diagnostique permettent de vérifier si le support IBA fonctionne correctement. Quelques exemples :

```
$ /usr/bin/ibv_devinfo

hca_id: mthca0
     fw_ver:                 3.5.0
     node_guid:                0002:c902:0027:6114
     sys_image_guid:             0002:c902:0027:6117
     vendor_id:               0x02c9
     vendor_part_id:            23108
     hw_ver:                 0xA1
     board_id:               MT_0270120001
     phys_port_cnt:             2
          port:  1
               state:              PORT_ACTIVE (4)
               max_mtu:              2048 (4)
               active_mtu:           2048 (4)
          port:  2
               state:              PORT_DOWN (2)
               max_mtu:              2048 (4)
               active_mtu:           512 (2)
```

Explications :

Le périphérique HCA Mellanox est reconnu, le nom du driver sous linux est "mthca0"

La version du firmware est 3.5.0 (il s'agit de la dernière disponible, voir "http://www.mellanox.com rubrique firmware"). Par conséquent, il n'y a pas lieu d'updater le firmware, en passant par l'utilitaire "mft-2"

Le port 1 est noté actif (PORT_ACTIVE) (un lien IBA est branché dessus).

La commande ci-après me renseigne sur la manière dont a été configuré le support IBA :

```
$ /etc/infiniband/info

prefix=/usr
Kernel=2.6.23.1-42.fc8
Configure options: --with-core-mod --with-user_mad-mod --with-user_access-mod --with-addr_trans-mod --with-mthca-mod --with-mlx4-mod --with-cxgb3-mod --with-nes-mod --with-ipath_inf-mod --with-ipoib-mod --with-sdp-mod --with-srp-mod --with-srp-target-mod --with-rds-mod
```

La commande ci-après me renseigne sur les modules et utilitaires installés

```
$ /usr/bin/ofed_info

OFED-1.3-mlnx1
libibverbs:
git://git.openfabrics.org/ofed_1_3/libibverbs.git ofed_1_3
commit 40b771aa6a9c0ad092b2e20775b4723d3b173792
libmthca:
git://git.openfabrics.org/ofed_1_3/libmthca.git ofed_1_3
commit 9501e698d257949acfab2edc90812602966dbcc9
libmlx4:
git://git.openfabrics.org/ofed_1_3/libmlx4.git ofed_1_3
commit 3869d6dab7e12fe452270ca641f7dd7082b42482
libehca:
git://git.openfabrics.org/ofed_1_3/libehca.git ofed_1_3
commit fd898180cfa3b737f893f432a80b91bac3396325
libipathverbs:
git://git.openfabrics.org/ofed_1_3/libipathverbs.git ofed_1_3
commit 82be4d81859d1fd2edf830220fe65a9923b80a46
libcxgb3:
git://git.openfabrics.org/ofed_1_3/libcxgb3.git ofed_1_3
commit 8218482d03898f859e1fa272649fafc64f91f05f
libnes:
git://git.openfabrics.org/ofed_1_3/libnes.git ofed_1_3
commit 471fa2e5a7bb2f8946119396358c31adcc6c2fb3
libibcm:
git://git.openfabrics.org/ofed_1_3/libibcm.git ofed_1_3
commit 53ec35f544bbc1838bbadc2210909c25a954a5e2
librdmacm:
git://git.openfabrics.org/ofed_1_3/librdmacm.git ofed_1_3
commit e7c55c78f8a8537966feba3ae61bcd5f043cbf97
dapl1:
git://git.openfabrics.org/ofed_1_3/dapl1.git ofed_1_3
commit 93eafe464227375d162e1fc764c2df5eab1ad50f
dapl2:
git://git.openfabrics.org/ofed_1_3/dapl2.git ofed_1_3
commit ee46aa47e2a4deeded347fe18bd6321db61c6594
libsdp:
git://git.openfabrics.org/ofed_1_3/libsdp.git ofed_1_3
commit c8102dccc502930442b23de658674d386456b350
sdpnetstat:
git://git.openfabrics.org/ofed_1_3/sdpnetstat.git ofed_1_3
commit 3341620a7259c4f7bdd4180864b98e260c3dc223
srptools:
git://git.openfabrics.org/ofed_1_3/srptools.git ofed_1_3
commit e0ce2d42eeb25f8e89b8f6daaa32a630c9b64f0d
```

```
perftest:
git://git.openfabrics.org/ofed_1_3/perftest.git ofed_1_3
commit 398da3f1d503b24141afa990fd46eb55c42a11cf
qlvnictools:
git://git.openfabrics.org/ofed_1_3/qlvnictools.git ofed_1_3
commit 086f9cb80ee790d61bddaf201ecbae32a2ff21dd
tvflash:
git://git.openfabrics.org/ofed_1_3/tvflash.git ofed_1_3
commit f5e7407a7f2058448df5e5320d9843f944427429
mstflint:
git://git.openfabrics.org/ofed_1_3/mstflint.git ofed_1_3
commit 260ae655d80fa3ed15c4544519a4c7f0cf768bd6
qperf:
git://git.openfabrics.org/ofed_1_3/qperf.git ofed_1_3
commit 6221aabd038df0b7033e035378ca190641ed2295
management:
git://git.openfabrics.org/ofed_1_3/management.git ofed_1_3
commit 2a37e8eb439553e448bdcf9e5905d5673d3b2820
ibutils:
git://git.openfabrics.org/ofed_1_3/ibutils.git ofed_1_3
commit 4dc546a0f37eb0651a7a66971ddb35531840301c
ibsim:
git://git.openfabrics.org/ofed_1_3/ibsim.git ofed_1_3
commit 55113d9f919709c7c97ea41d29991941b9c8be70

ofa_kernel-1.3:
Git:
git://git.openfabrics.org/ofed_1_3/linux-2.6.git ofed_kernel
commit cdbdfc5cc29c4add1a2d6967b137a3347112a199

# MPI
mvapich-1.0.0-2106.src.rpm
mvapich2-1.0.2-4.src.rpm
openmpi-1.2.5-1.src.rpm
mpitests-3.0-773.src.rpm
```

Les scripts de démarrage associés au support IBA sont installés sous /etc/init.d/.

2 scripts IBA sont installés :

- **openibd** : script pour le chargement des modules IBA et drivers HCA (configuration sous /etc/infiniband/openib.conf)
- **opensmd** : démon openSM (avec son fichier de config sous "/etc/opensm.conf")

Pour éviter les erreurs résultant du chargement de modules inutiles car associés à des périphériques InfiniBand absents sur le nœud maître, on prend soin d'ajuster le fichier de configuration /etc/infiniband/openib.conf.

Par exemple, le démarrage de openibd aboutit aux erreurs suivantes :

```
$ /etc/init.d/openibd restart

Unloading HCA driver:                                      [  OK  ]
Loading Mellanox MLX4 HCA driver:                          [FAILED]
Loading QLogic InfiniPath driver:                          [FAILED]
Loading cxgb3 driver:                                      [FAILED]
Loading HCA driver and Access Layer:                       [FAILED]
Please open an issue in the http://openib.org/bugzilla and attach /tmp/ib_debug_info.log
```

Sur notre configuration matérielle, ces erreurs sont résolues en ajustant openib.conf de la manière suivante :

MLX4 driver : est le Mellanox MLX4 HCA driver pour carte ConnextX

on place ‘’**MLX4_LOAD=no’’** dans /etc/infiniband/openib.conf

QLogic InfiniPath driver : (Mellanox HCA driver pour carte QLogic InfiniPath)

on place ‘’**IPATH_LOAD=no’’** dans /etc/infiniband/openib.conf

cxgb3 : (User space library for the Chelsio T3 iWARP RNIC adapter. This library is a plug-in module for libibverbs, the Linux user space RDMA library)

on place ‘’**CXGB3_LOAD=no’’** dans /etc/infiniband/openib.conf

Après ces modifications et redémarrage des services, on vérifie l’issue du chargement des modules IBA et des drivers HCA :

```
$ /etc/init.d/openibd start

$ /etc/init.d/openibd status

HCA driver loaded
Configured devices:
ib0 ib1
Currently active devices:
ib0
ib1
The following OFED modules are loaded:
 rdma_ucm
 rdma_cm
 ib_addr
 ib_ipoib
 ib_mthca
 ib_uverbs
 ib_umad
 ib_sa
 ib_cm
 ib_mad
 ib_core
```

Dans la foulée, on vérifie que le démon openSMD démarre correctement et que le matériel IBA est bien détecté au moyen du log système :

```
$ /etc/init.d/opensmd start

Starting IB Subnet Manager.                         [  OK  ]

$ dmesg

ib_mthca: Mellanox InfiniBand HCA driver v0.08 (February 14, 2006)
ib_mthca: Initializing 0000:04:00.0
ib_mthca: Mellanox InfiniBand HCA driver v0.08 (February 14, 2006)
ib_mthca: Initializing 0000:04:00.0
ADDRCONF(NETDEV_UP): ib0: link is not ready
ADDRCONF(NETDEV_UP): ib1: link is not ready
ADDRCONF(NETDEV_CHANGE): ib0: link becomes ready
ib0: no IPv6 routers present
```

L'état retourné par l'utilitaire sminfo permet de vérifier si le gestionnaire Open Subnet Manager détecte de l'activité :

```
$ /usr/sbin/sminfo

sminfo: sm lid 1 sm guid 0x2c90200276115, activity count 145 priority 0 state 3
SMINFO_MASTER
```

openSMD détecte de l'activité, dans le cas contraire (service arrêté ou problème matériel), sminfo retourne l'état : ''SMINFO_STANDBY'' (pas d'activité détecté) ou ''iberror : query failed'' (service arrêté).

Les autres utilitaires de diagnostique particulièrement utiles sont :

ibhosts - ibdiagnet - ibchecknet - ibnetdiscover - ibswitches

- **Etape 4 : Installation du support IBA sur les noeuds de calcul**

Le déploiement sur les nœuds est réalisé en suivant la même procédure que pour le nœud maître, en exécutant sur chaque nœud le script **install.pl**. Le fichier de configuration ''ofed.conf'' (cf Annexe E, script ''ofed_nodes.conf'') est ajusté de manière à éviter l'installation des modules superflus (car associés à des périphériques IBA absents), supprimer l'installation sur les nœuds des services d'administration redondants (openSM, IBadm, etc ..) et éviter l'installation des librairies MPI spécialement compilés avec le support InfiniBand.

Sur chaque nœud, comme précédemment, l'issue du processus d'installation est vérifiée avec les utilitaires de diagnostique, après avoir relancé le script de démarrage **openibd** sur chaque machine.

Enfin, sur le nœud maître, le service **opensmd** est relancé. La vérification avec ibchecknet permet de s'assurer que l'ensemble du réseau IBA est correctement détecté à travers le cluster. Sur notre cluster, composé de 10 serveurs et 1 switch InfiniScale, l'issue correcte est :

```
$ ibchecknet

## Summary: 11 nodes checked, 0 bad nodes found
##          20 ports checked, 0 bad ports found
##          1 ports have errors beyond threshold
```

Explication : 11 nodes détectés, en référence aux 10 HCA InfiniHost + 1 switch InfiniScale.

5. Installation de la librairie openMPI avec support InfiniBand

Avec nos différents codes parallélisés, nous utilisons avec succès la librairie openMPI. Plusieurs librairies MPI ont été testées, parmi lesquelles MPI Intel, MPICH, LAM …, mais le choix de l'administrateur porte sur openMPI, pour les raisons suivantes :

- avec MPI Intel, la librairie openMPI est l'une des plus performante
- elle est openSource, contrairement à MPI Intel
- la documentation, tutoriels, le support (forum) et la réactivité de l'équipe de développement Open MPI est excellente

Installation d'openMPI

Pour son installation, le lecteur est invité à se rendre sur le site http://www.open-mpi.org et vérifier la dernière version disponible en suivant le lien : http://www.open-mpi.org/software

Les fichiers INSTALL.txt et README.txt inclus dans le paquetage openMPI présentent toutes les informations nécessaires à l'installation de la librairie. Enfin, la page ''Faq'' (http://www.open-mpi.org/faq/) constitue une mine d'informations sur les étapes d'installation, de démarrage, débogage et les pistes d'optimisation, etc.

On donne ici la configuration d'openMPI sur le cluster et les points essentiels à vérifier :

```
$ ompi_info

Open MPI: 1.2.6
Open MPI SVN revision: r17946
Open RTE: 1.2.6
Open RTE SVN revision: r17946
OPAL: 1.2.6
OPAL SVN revision: r17946
Prefix: /opt/openmpi/openmpi-1.2.6_intel
Configured architecture: x86_64-unknown-linux-gnu
Configured by: root
Configured on: Fri May 30 15:17:37 NCT 2008
Configure host: editr.cluster.ird.nc
Built by: root
Built on: Fri May 30 15:31:37 NCT 2008
Built host: editr.cluster.ird.nc
C bindings: yes
C++ bindings: yes
Fortran77 bindings: yes (all)
Fortran90 bindings: yes
Fortran90 bindings size: small
C compiler: icc
C compiler absolute: /opt/intel/cce/10.1.015/bin/icc
C++ compiler: icpc
C++ compiler absolute: /opt/intel/cce/10.1.015/bin/icpc
Fortran77 compiler: ifort
Fortran77 compiler abs: /opt/intel/fce/10.1.015/bin/ifort
Fortran90 compiler: ifort
Fortran90 compiler abs: /opt/intel/fce/10.1.015/bin/ifort
C profiling: yes
C++ profiling: yes
Fortran77 profiling: yes
Fortran90 profiling: yes
C++ exceptions: no
```

```
Thread support: posix (mpi: no, progress: no)
Internal debug support: no
MPI parameter check: runtime
Memory profiling support: no
Memory debugging support: no
libltdl support: yes
Heterogeneous support: yes
mpirun default --prefix: no
MCA backtrace: execinfo (MCA v1.0, API v1.0, Component v1.2.6)
MCA memory: ptmalloc2 (MCA v1.0, API v1.0, Component v1.2.6)
MCA paffinity: linux (MCA v1.0, API v1.0, Component v1.2.6)
MCA maffinity: first_use (MCA v1.0, API v1.0, Component v1.2.6)
MCA maffinity: libnuma (MCA v1.0, API v1.0, Component v1.2.6)
MCA timer: linux (MCA v1.0, API v1.0, Component v1.2.6)
MCA installdirs: env (MCA v1.0, API v1.0, Component v1.2.6)
MCA installdirs: config (MCA v1.0, API v1.0, Component v1.2.6)
MCA allocator: basic (MCA v1.0, API v1.0, Component v1.0)
MCA allocator: bucket (MCA v1.0, API v1.0, Component v1.0)
MCA coll: basic (MCA v1.0, API v1.0, Component v1.2.6)
MCA coll: self (MCA v1.0, API v1.0, Component v1.2.6)
MCA coll: sm (MCA v1.0, API v1.0, Component v1.2.6)
MCA coll: tuned (MCA v1.0, API v1.0, Component v1.2.6)
MCA io: romio (MCA v1.0, API v1.0, Component v1.2.6)
MCA mpool: rdma (MCA v1.0, API v1.0, Component v1.2.6)
MCA mpool: sm (MCA v1.0, API v1.0, Component v1.2.6)
MCA pml: cm (MCA v1.0, API v1.0, Component v1.2.6)
MCA pml: ob1 (MCA v1.0, API v1.0, Component v1.2.6)
MCA bml: r2 (MCA v1.0, API v1.0, Component v1.2.6)
MCA rcache: vma (MCA v1.0, API v1.0, Component v1.2.6)
MCA btl: self (MCA v1.0, API v1.0.1, Component v1.2.6)
MCA btl: sm (MCA v1.0, API v1.0.1, Component v1.2.6)
MCA btl: openib (MCA v1.0, API v1.0.1, Component v1.2.6)
MCA btl: tcp (MCA v1.0, API v1.0.1, Component v1.0)
MCA topo: unity (MCA v1.0, API v1.0, Component v1.2.6)
MCA osc: pt2pt (MCA v1.0, API v1.0, Component v1.2.6)
MCA errmgr: hnp (MCA v1.0, API v1.3, Component v1.2.6)
MCA errmgr: orted (MCA v1.0, API v1.3, Component v1.2.6)
MCA errmgr: proxy (MCA v1.0, API v1.3, Component v1.2.6)
MCA gpr: null (MCA v1.0, API v1.0, Component v1.2.6)
MCA gpr: proxy (MCA v1.0, API v1.0, Component v1.2.6)
MCA gpr: replica (MCA v1.0, API v1.0, Component v1.2.6)
MCA iof: proxy (MCA v1.0, API v1.0, Component v1.2.6)
MCA iof: svc (MCA v1.0, API v1.0, Component v1.2.6)
MCA ns: proxy (MCA v1.0, API v2.0, Component v1.2.6)
MCA ns: replica (MCA v1.0, API v2.0, Component v1.2.6)
MCA oob: tcp (MCA v1.0, API v1.0, Component v1.0)
MCA ras: dash_host (MCA v1.0, API v1.3, Component v1.2.6)
MCA ras: localhost (MCA v1.0, API v1.3, Component v1.2.6)
MCA ras: gridengine (MCA v1.0, API v1.3, Component v1.2.6)
MCA ras: slurm (MCA v1.0, API v1.3, Component v1.2.6)
MCA ras: tm (MCA v1.0, API v1.3, Component v1.2.6)
MCA rds: hostfile (MCA v1.0, API v1.3, Component v1.2.6)
MCA rds: proxy (MCA v1.0, API v1.3, Component v1.2.6)
MCA rds: resfile (MCA v1.0, API v1.3, Component v1.2.6)
MCA rmaps: round_robin (MCA v1.0, API v1.3, Component v1.2.6)
MCA rmgr: proxy (MCA v1.0, API v2.0, Component v1.2.6)
MCA rmgr: urm (MCA v1.0, API v2.0, Component v1.2.6)
MCA rml: oob (MCA v1.0, API v1.0, Component v1.2.6)
MCA pls: proxy (MCA v1.0, API v1.3, Component v1.2.6)
```

```
MCA pls: gridengine (MCA v1.0, API v1.3, Component v1.2.6)
MCA pls: rsh (MCA v1.0, API v1.3, Component v1.2.6)
MCA pls: slurm (MCA v1.0, API v1.3, Component v1.2.6)
MCA pls: tm (MCA v1.0, API v1.3, Component v1.2.6)
MCA sds: env (MCA v1.0, API v1.0, Component v1.2.6)
MCA sds: seed (MCA v1.0, API v1.0, Component v1.2.6)
MCA sds: singleton (MCA v1.0, API v1.0, Component v1.2.6)
MCA sds: pipe (MCA v1.0, API v1.0, Component v1.2.6)
MCA sds: slurm (MCA v1.0, API v1.0, Component v1.2.6)
```

Cette version d'openMPI est couplée avec le compilateur Intel, pour le support CC, C++ et Fortran :

```
C compiler: icc
C++ compiler: icpc
Fortran77 compiler: ifort
```

On vérifiera que l'affinité des processus/thread au processeur est intégrée dans la version de la librairie MPI (voir http://www.open-mpi.org/faq/?category=tuning#paffinity-defs) :

```
MCA paffinity: linux (MCA v1.0, API v1.0, Component v1.2.6)        OK
MCA maffinity: first_use (MCA v1.0, API v1.0, Component v1.2.6)    OK
MCA maffinity: libnuma (MCA v1.0, API v1.0, Component v1.2.6)      OK
```

De même, on prendra soin de vérifier que les différentes couches de transfert des données (BTL = Byte Transfert Layer) disponibles sur le cluster sont supportées dans la version openMPI installée. Ici les protocoles détectés/supportés sont openFabrics (InfiniBand), TCP à travers le réseau GigE, SM (Shared Memory) et Loopback (Send-to-Self)

```
MCA btl: self (MCA v1.0, API v1.0.1, Component v1.2.6)
MCA btl: sm (MCA v1.0, API v1.0.1, Component v1.2.6)
MCA btl: openib (MCA v1.0, API v1.0.1, Component v1.2.6)
MCA btl: tcp (MCA v1.0, API v1.0.1, Component v1.0)
```

Enfin, la librairie openMPI supporte l'exécution des jobs au moyen de l'interface Open PBS (le gestionnaire de job venant avec le Cluster Toolkit Oscar)

```
MCA pls: tm (MCA v1.0, API v1.3, Component v1.2.6)
```

Exécution des jobs et sélection du réseau et du protocole de communication

A l'évocation du script ''mpirun'' pour l'exécution du job parallélisé, on peut spécifier d'utiliser l'une ou l'autre des couches de transfert, par exemple :

Pour utiliser exclusivement les échanges via le réseau GigE par TCP :

```
$ mpirun –mca btl tcp,self –np 6  roms.exe roms.in
```

Pour utiliser exclusivement les échanges via le réseau InfiniBand :

```
$ mpirun –mca btl self,openib,^tcp –np 6 roms.exe roms.in
```

En pratique sur notre architecture, l'évocation simplifiée du lanceur de job, par exemple ''mpirun –np 6 mon.exe'' suffit pour que l'échange des données emprunte l'InfiniBand, dans la mesure où par défaut, le réseau le moins chargé et le plus rapide est automatiquement sélectionné.

6. Post-Installation : autres paramètres système à ajuster

Dans l'environnement Unix/Linux, les paramètres de système sont définis par défaut pour permettre de bloquer uniquement une quantité minimum de mémoire par les applications en mode utilisateur.

OpenFabrics nécessite de dépasser cette limite et dans l'idéal il est conseillé de ne pas la fixer (valeur = unlimited) (Cf http://www.open-mpi.org/faq/?category=openfabrics#ib-locked-pages). Dans le cas contraire les jobs ne partent pas et un message d'erreur s'affiche du type :

```
libibverbs: Warning: RLIMIT_MEMLOCK is 32768 bytes.
   This will severely limit memory registrations.
```

Pour augmenter cette limite, il est nécessaire de modifier les paramètres dans ''/etc/security/limits.conf'', en y ajoutant les lignes suivantes :

```
*               soft    memlock         unlimited
*               hard    memlock         unlimited
```

7. Gains apportés par l'InfiniBand

Dans cette section, on se propose de présenter les tests comparatifs de vitesse d'exécution des jobs avec les codes ROMS, WRF et MARS parallélisés. Le compilateur Intel et la librairie openMPI est utilisée pour ces tests. Les options de compilation passées au compilateur Ifort ne sont pas modifiées, seul le réseau/protocole de communication est changé. Au moyen des arguments passés au script d'exécution ''mpirun'', il est possible de spécifier le réseau utilisé pour l'échange des données :

- **-- mca btl tcp,self** pour transfert par le réseau GigE
- **-- mca btl self,openib** pour transfert par le réseau InfiniBand

Configuration ROMS utilisée

La configuration Safe de Pierrick Penven utilisée précédemment pour les benchmarks sur le cluster de Nouméa et Cape Town (voir Projet Spirales 2008, Annexe C) est de nouveau employée pour le test de performance.

Il s'agit d'une configuration ROMS sur l'Afrique du Sud, avec comme grille 228 X 192 X 32. Le modèle est intégré sur 1 mois, soit 1200 itérations. Cette configuration n'emploi pas d'emboîtement.

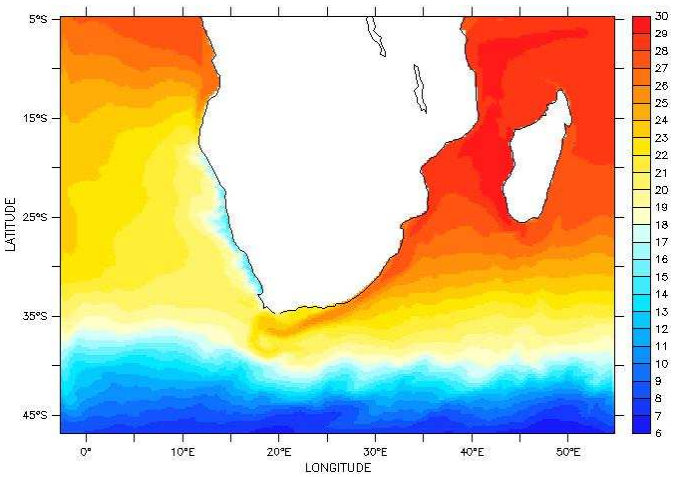


Figure 4 Emprise géographique de la configuration ROMS (Safe), représentée ici par le champs de température en surface

Configuration WRF utilisée

Une configuration régionale centrée sur la Nouvelle-Calédonie de WRF est également employée pour le test de performance.

La configuration emploi 2 domaines avec emboîtement online et two-way. Le domaine 1 est une grille 59 X 59 X 60 et le domaine 2 une grille 165 X 165 X 60. Le modèle est intégré pendant 48 heures.

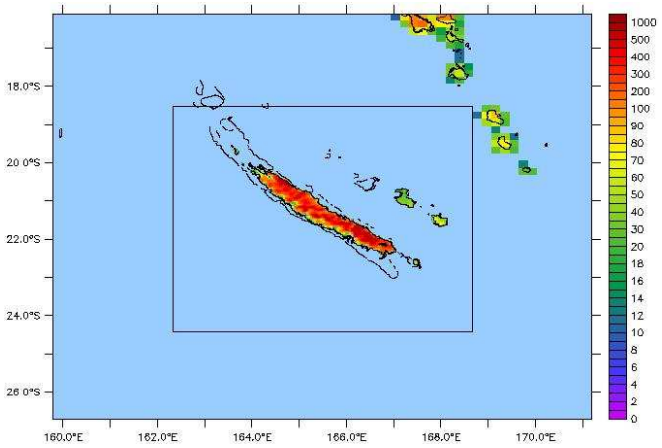


Figure 5 Emprise géographique de la configuration régionale WRF, représentée ici par la topographie (mètres). Le rectangle intérieur symbolise l'emprise du second domaine emboité.

Configuration MARS utilisée

La configuration régionale actuellement employée dans le cadre du projet HydroPerli (http://hydro-perli.ird.nc/), est utilisée pour le test de performance. Le domaine est centré sur l'archipel des *Tuamotu* (Polynésie Française).

La configuration emploi le code MARS dans sa version 2D pour le calcul de la marée sur une grille 884 X 445.

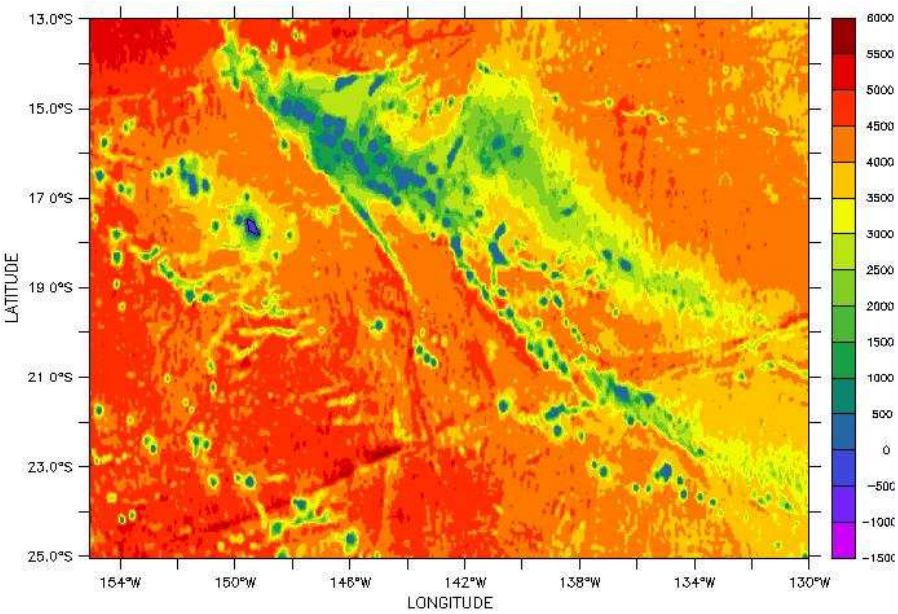


Figure 6 Emprise géographique de la configuration régionale MARS, représentée ici par la topographie (mètres).

Résultats avec le code ROMS (Safe)

En premier constat, la technologie InfiniBand permet de réduire considérablement l'overhead des processeurs, ce qui conduit à diminuer sensiblement le temps de calcul.

La Figure 7 illustre l'utilisation moyenne des 20 processeurs mobilisés pour le calcul avec la configuration ROMS Safe. En employant la couche de transfert GigE TCP/IP, on constate que 50% du temps de calcul est dépensé inutilement en raison des interruptions de CPU provoquées par la gestion des échanges.

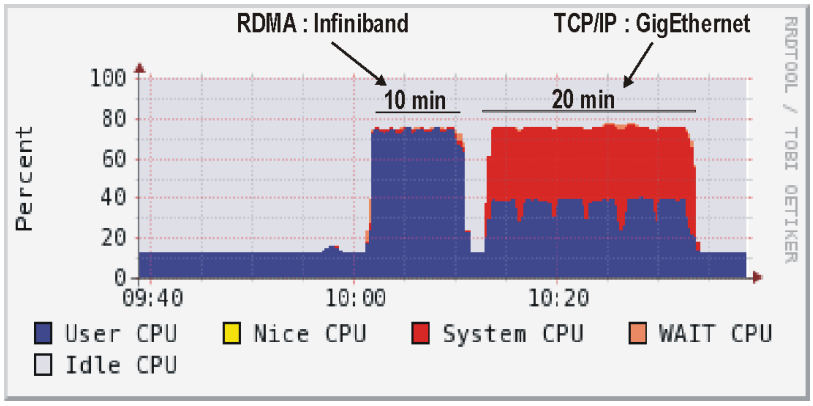


Figure 7 Illustration de l'Overhead des 20 processeurs pour la configuration ROMS (Safe) avec à gauche, l'échange par réseau InfiniBand et à droite l'échange par réseau GigE

Dans cette configuration à 20 processeurs, l'architecture InfiniBand a permis de réduire le temps de calcul par un facteur 2.

Pour situer le cluster de Nouméa par rapport au cluster de Cape Town constitué de processeurs Opteron 2218 et équipé d'un réseau InfiniBand et GigE, les résultats des précédents tests réalisés par Pierrick Penven avec ROMS Safe sont repris et comparés. Les deux clusters sont composés de machines bi-processeur à double-cœur comparables.

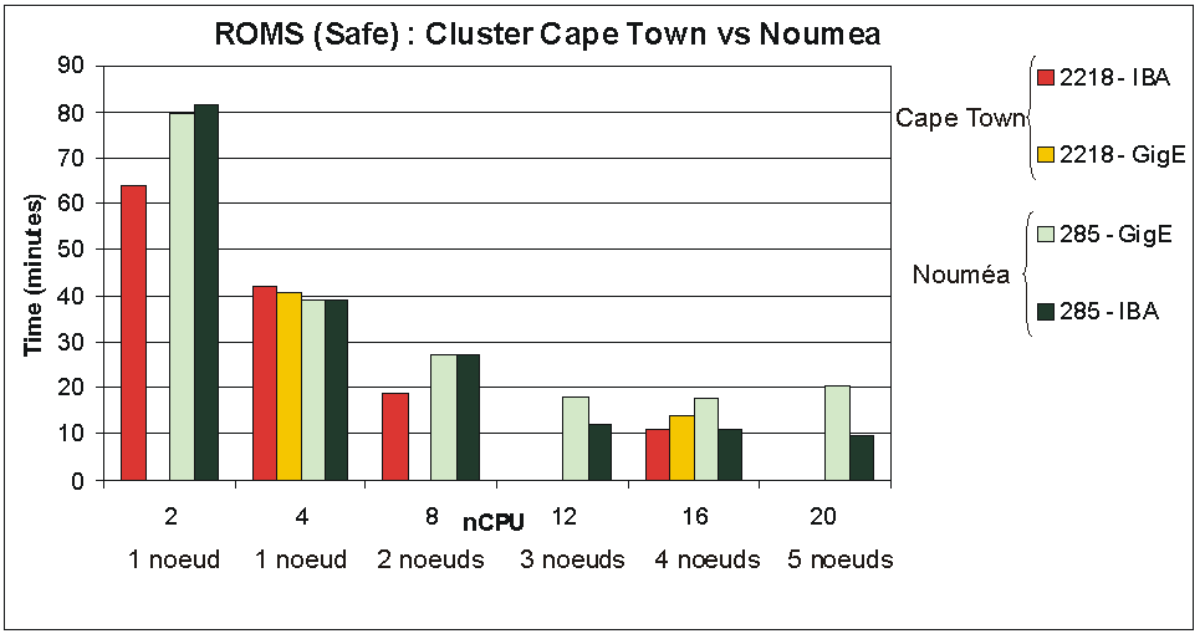


Figure 8 Comparaison des temps de calcul pour la configuration ROMS Safe selon différents modes de partitionnement et modes de transfert entre l'architecture de Cape Town et celle de Nouméa.

La Figure 8 montre qu'à partir de 3 nœuds de calcul, la performance du cluster de Nouméa atteint celle du cluster de cape Town. Toutefois, pour un partitionnement impliquant 2

machines de calcul, l'apport de l'IBA sur le GigE n'est pas évident. L'apport de l'IBA est sensible à partir d'un partitionnement impliquant 12 CPUs (3 machines de calcul).

Les courbes de Speed-up[3] (Cf Figure 9) permettent de vérifier comment la nouvelle architecture équipée InfiniBand est scalable pour le code ROMS (Safe). Avec le transfert GigE TCP, la capacité du code ROMS à s'accélérer avec la montée en puissance du nombre de CPU(s) s'effondre très rapidement. Au contraire, le code ROMS montre une bonne capacité à profiter de l'augmentation du nombre de processeurs dans une architecture équipée du réseau InfiniBand. Sur le cluster de Nouméa, l'évolution n'est cependant pas totalement linéaire. Sur une architecture de calcul équilibrée, le code ROMS montre une bonne scalabilité.

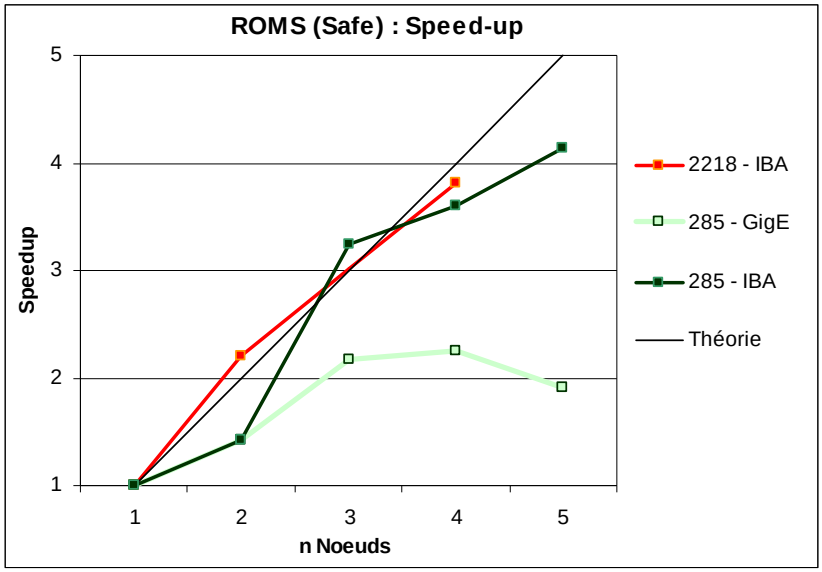


Figure 9 Courbes de Speed-Up du code ROMS (Safe) sur l'architecture IBA Cape Town (rouge) et l'architecture IBA et GigE de Nouméa

Remarque :

Nous avons conservé le test avec la configuration Safe, pour permettre la comparaison avec le cluster de Cape Town. Néanmoins, pour de futurs benchmarks basés sur ROMS il est conseillé de définir une configuration régionale en pleine eau, sans bande terrestre, de manière à éviter les déséquilibres de charge entre processeurs affectés aux calculs des équations en pleine eau et ceux affectés aux calculs sur les domaines terrestres. Ces déséquilibres peuvent par exemple expliquer en partie le comportement non linéaire des performances constaté sur le cluster de Nouméa (Cf Figure 9).

Résultats avec le code WRF

Cette configuration induit des calculs lourds, en grande partie dominés par le domaine emboîté (165 X 165 X 60). Comme le montre la Figure 10, entre une configuration du modèle employant respectivement 12 puis 24 CPUs, le transfert par GigE TCP est

[3] Le Speed-up est défini par S(n) = T(1)/T(n), avec T(n) le temps d'exécution (Wall Time)

dramatiquement pénalisant et ne permet pas de profiter de la scalabilité du code WRF. Au contraire, en employant l'échange par IBA, le code WRF profite de l'augmentation du nombre de processeurs alloués. Par exemple, le doublement du nombre de CPUs s'accompagne d'un facteur de Speed-Up de 1.73 et pour un partitionnement impliquant 24 CPUs, le temps de calcul est divisé par un facteur 3 entre l'architecture GigE et IBA.

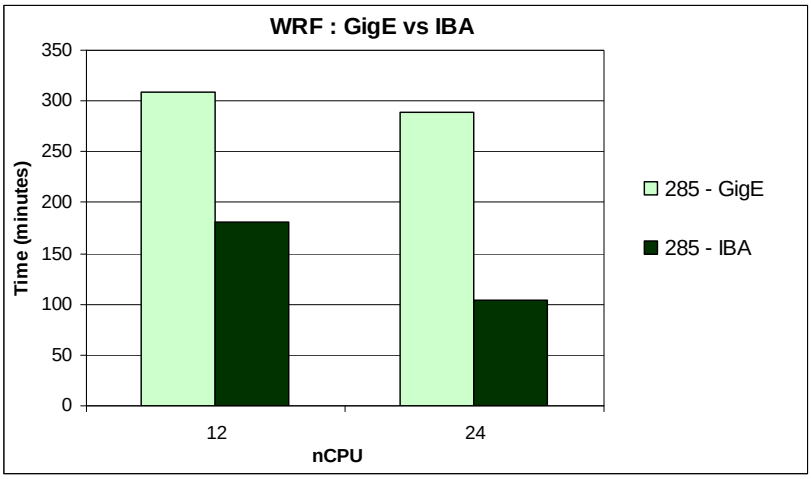


Figure 10 Comparaison des temps d'exécution pour la configuration WRF selon différents modes de partitionnement et pour les deux modes d'échanges GigE TCP et InfiniBand

Résultats avec le code MARS

Les courbes de Speed-up (Cf Figure 11) permettent de vérifier la scalabilité du code MARS dans sa version 2D selon le réseau de communication employé.

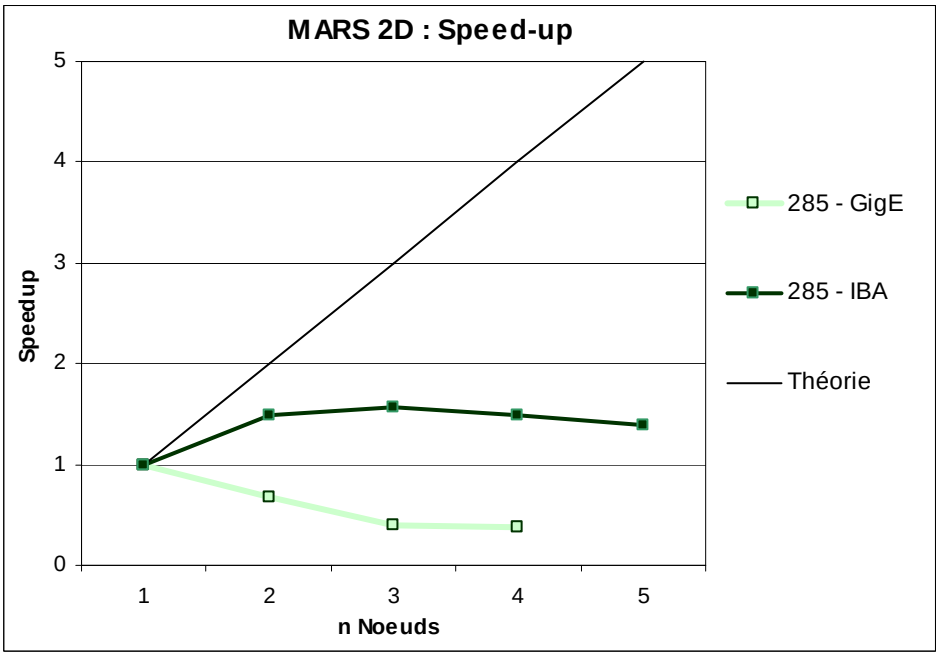


Figure 11 Courbes de Speed-Up du code MARS 2D sur l'architecture IBA et GigE de Nouméa

Comparativement au transfert GigE, l'InfiniBand permet d'améliorer nettement les temps d'exécution, mais la stratégie de parallélisation MPI mise en œuvre dans le code de calcul

ne permet pas de profiter pleinement de la montée en puissance du nombre de processeurs. Dans cette configuration, au-delà de 8 CPUs, la vitesse d'exécution du code n'augmente pas, au contraire. Force est de constater que sur une architecture à mémoire distribuée employant un réseau de communication rapide, le schéma temporel implicite employé dans MARS rend l'efficacité de sa parallélisation MPI très délicate. Dans sa version 3D, les gains obtenus sur l'architecture InfiniBand seront probablement plus visibles.

8. Pistes d’optimisation de l’architecture de calcul de Nouméa

Avec l’upgrade des processeurs Opterons version 246 (monocoeur) vers la version 285 (bicoeur), puis la récente installation du réseau InfiniBand, les possibilités hardware d’optimisation du cluster sont désormais atteintes. L’architecture est aujourd’hui équilibrée.

Pour répondre à une demande croissante en calcul, le cluster de Nouméa offre la possibilité d’ajouter des nœuds de calcul dans la grappe, comme ceci avait été effectué en 2007. L’IBA installé et son switch Flextronics DDR 20 Gb/s 24 ports permettent d’accueillir des machines de calcul modernes équipées multi-cœur et slots PCI-E, lesquels pourront être exploitées de manière optimale, sans limitation imposée par le réseau.

Il existe cependant des pistes logiciels d’optimisation, qui peuvent faire l’objet de demande de soutien de la part de la DSI, et/ou au travers le SIL de Nouméa, par exemple :

- L’installation sur le cluster d’une solution de montage des fichiers à travers le cluster par NFS-RDMA, afin de permettre les opérations de lecture et d’écriture à travers RDMA plutôt que par NFS-UDP comme c’est actuellement le cas. Cette opération peut s’avérer très payante pour les configurations gourmandes en opérations E/S (par exemple MARS et WRF).
- L’optimisation des codes de calcul (exemple parallélisation hybride du code ROMS, stratégie de parallélisation du code MARS)

ANNEXE A Présentation des activités de modélisation en grande partie réalisées sur le cluster de Nouméa et exemple de portage de l'expérience vers le Pérou

(source : Lettre Mercator n°30 - Juillet 2008

http://www.mercator-ocean.fr/html/lettre/presentation_lettre_fr.html)

Keys to affordable regional marine forecast systems

By Patrick Marchesiello[1], Jérôme Lefèvre[1], Pierrick Penven[2], Florian Lemarié[1], Laurent Debreu[3], Pascal Douillet[1], Andres Vega[1], Patricia Derex[1], Vincent Echevin[4] and Boris Dewitte[5]

[1] IRD/LEGOS/UR065, IRD Noumea, Centre BPA05, 98848 Noumea Cedex, Nouvelle Caledonie.
[2] IRD/UR097, ECO-UP, Centre IRD de Bretagne, B.P. 70, 29280 Plouzane, France
[3] INRIA, 51 rue des mathémathiques, B.P. 53, 38041 Grenoble, France
[4] IRD/LOCEAN, Institut Pierre Simon Laplace, Boîte 100 - 4, place Jussieu 75252 PARIS Cedex 05, France
[5] IRD/LEGOS/IMARPE, Esquina de Gamarra y General Valle S/N Chucuito, Callao, Peru

Introduction

Only the wealthiest countries in the world can propose governmental services with decent financial support to provide marine forecasting on an operational level. If a laboratory of oceanography or meteorology in a developing country needs to provide marine forecast to support its country's economy or prevent pollution, *what alternative can be found*? This is precisely the kind of questions that the Institute of Research for Development (IRD) is designed to answer.

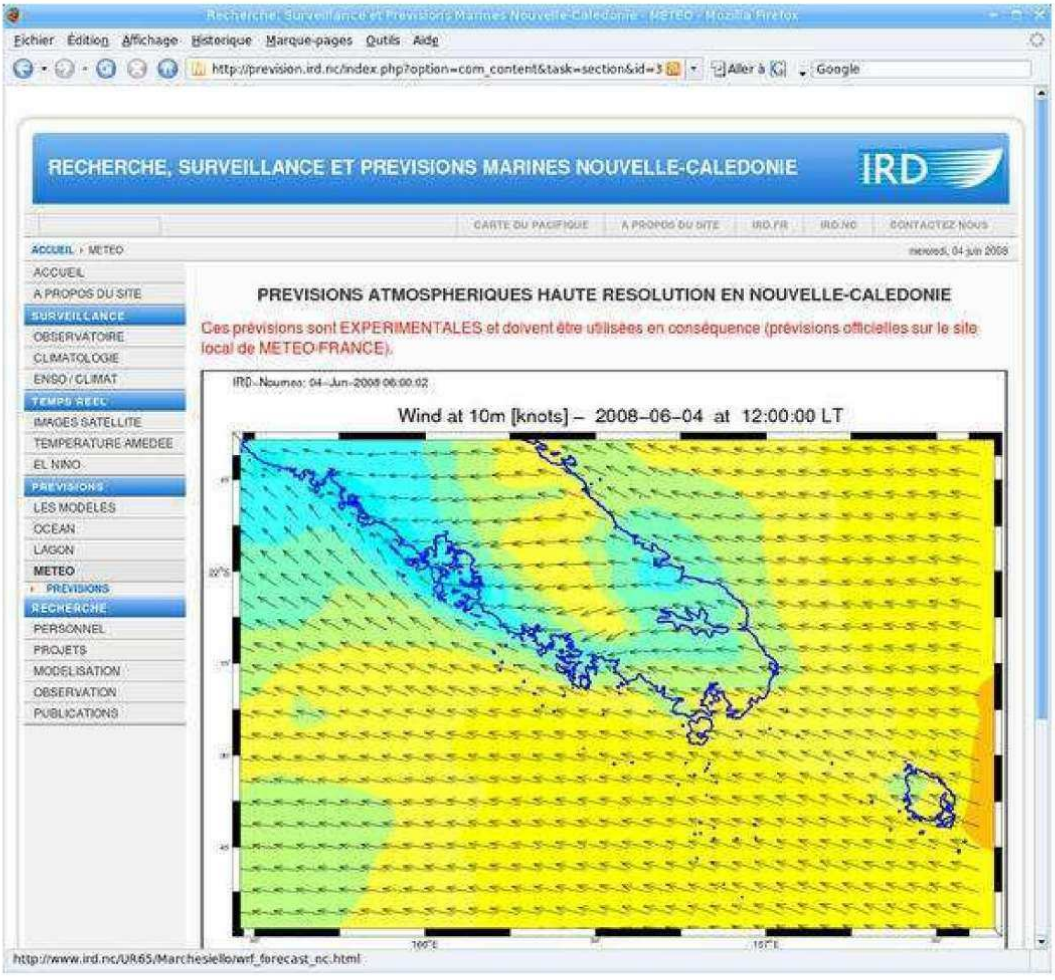


Figure 1

The oceanographic research group at the Nouméa-IRD center proposes a functional but experimental nowcast/forecast integrated system of marine meteorology, deep sea and lagoon oceanography which is updated everyday through the web site: http://prevision.ird.nc

Our understanding is that a regional marine forecast system needs at its core some state-of-the-art regional oceanic and atmospheric models which can be applied in a downscaling mode to refine the results of existing global systems, rather than generating entirely new solutions. The challenge is to find *freely available* models and global operational data sets, then build interfaces extracting these data, interpolate it to the regional grids, integrate the models in both nowcast and forecast modes, analyze the results and present them to potential users through a distributed media (the World Wide Web). A key condition for our system to work is that maintenance be minimal, which can only be achieved if the system is entirely *automated* through scheduled commands and scripts.

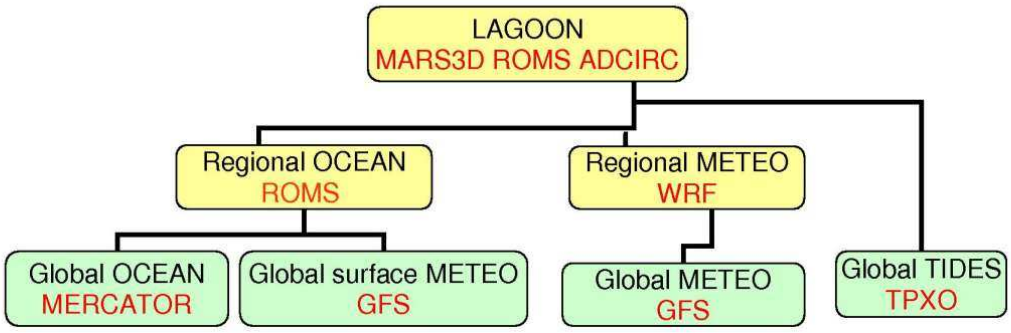


Figure 2

Diagram presenting the different parts of the IRD integrated operational system. Extraction and transfer from global (green) to regional (yellow) stages are done using the OPeNDAP extraction protocol; each step is completely automated through the Linux CRON service. It takes about 2 hours every night for the entire system to run on 4 processors.

With support from the PREVIMER project, as well as IRD (Institut de Recherche pour le Développement), ANR (Agence Nationale de la Recherche) and ZoNéCo (New Caledonia government research program), an experimental but fully automated system was created for marine survey and forecast of the New-Caledonia region (http://prevision.ird.nc ; Figure 1). New Caledonia is well suited for the experiment because it gathers most of the constraints encountered in developing countries, i.e. slow internet connection, remoteness from data centers, expensive or unavailable hardware and software. Meanwhile, good research conditions at the IRD-Nouméa center and a comfortable financial support has provided the means to explore different paths and reach the experiment's goal. In a second phase, the experimental system should be exported towards developing countries. This step is underway as parts of the system are already active in Peru, Mexico, Senegal and South Africa.

The marine forecast system of New-Caledonia is rather complex as it comprises 3 parts: the oceanic, the lagoon and the atmospheric models (Figure 2). In the following we present those different parts, emphasizing key points that are relevant to a portable and affordable operational system.

Hardware

Recent modeling developments have greatly benefited from the rapid progress of Personal Computers, including hardware, operating systems, libraries and compilers. Those in turn are largely profiting from the free software era in which we now evolve. As a result, multiple processor machines are so efficient that single-core, single-processor machines are not found on the market anymore. Therefore, oceanic and atmospheric modeling is unconceivable without code parallelization, which can simultaneously use multiple cores of one processor, or/and multiple processors of one PC, or/and multiple PCs of a cluster. Under these circumstances and because our models are efficiently parallelized, we have found support during the last 3 years to progressively build an experimental PC cluster in Noumea for scientific computing (Figure 3); our objective is to demonstrate that a cheap and efficient local-size machine (40 processor units) can be built in developing countries. In this experiment, we have learned that all necessary computer software (operating system, cluster administration system, communication libraries, compilers, job manager system) is freely available and functional, provided that a few options are properly set. The performances of local-size PC clusters can be excellent, although it shows a few bottle necks such as dual-core parallelization (there is efficiency loss using message passing protocols on a dual-core processor compared to a bi-processor, but the loss has been found acceptable) and more importantly the PCs connection system. A high communication bandwidth capability appears indispensable when the number of grid points exchanging information becomes large compared to the total grid points (with slow connection, the volume of data transfer becomes too large compared to actual computing). Therefore, our choice went to the InfiniBand connection (10Gbps of bandwidth) which allows using the full cluster capacity even on computational grids that

are relatively small. Based on the lessons learned and selected choices, a similar PC cluster is now being implemented in Lima (Peru) as a second phase of the experiment.

Figure 3

Nouméa PC cluster, made of 10 bi-dual-core Opteron processor 246 and 285, with motherboard Tyan S2885 and Infiniband connection for communication between the PCs. The Cluster was assembled at the IRD-Nouméa center.

The oceanic system

This section only concerns the oceanic system at the regional scale (Figure 2). The lagoon of New Caledonia, which is an isolated body of water with limited connections to the ocean, is treated in the section below "Coupling".

Model and configurations

The Regional Ocean Modeling System (ROMS) is a new generation ocean circulation model (Shchepetkin and McWilliams, 2005). It has been specially designed for the accurate simulation of regional oceanic systems with coastal margins. IRD jointly with INRIA (Institut National de Recherche en Informatique et en Automatique) are maintaining a separate branch of the model (ROMS_AGRIF) which features multi-level 2-way nesting capability (Penven et al., 2006), and a modified code structure to straightforwardly handle realistic modeling and its application modules. The conjunction of a stable, accurate and efficient model with robust open boundary conditions (Marchesiello et al., 2001) allows us to follow a systematic approach to obtain a "standard" regional configuration (Marchesiello et al., 2003). To that end, a series of MATLAB routines and several essential datasets were gathered in an integrated toolbox called ROMSTOOLS (Penven et al., 2007; the toolbox and the model can be freely downloaded at: http://www.brest.ird.fr/Roms_tools/). ROMSTOOLS requires only a minimum list of elements (location and size of parent and nested domains, resolutions …) to obtain a model simulation for almost any region. The standard pre-processing tools include interpolation procedures, generic land masking and topography smoothing. Refinements are then added as needed, i.e. nested domains, more accurate wind forcing, initial, boundary and topographic data, as well as various biogeochemical data for relevant applications. In various complex configurations, ROMSTOOLS and ROMS_AGRIF have been applied successfully on a variety of regional studies of the world ocean (Blaas et al., 2007; Blanke et al., 2002; Capet et al., 2004; Karakas et al., 2006; Gruber et al., 2006; Marchesiello et al., 2003, 2007; Messie et al., 2006; Peliz et al., 2007ab; Penven et al., 2000, 2006; among others). Note that ROMSTOOLS should evolve in the future towards free programming languages replacing MATLAB (OCTAVE for example) in order to better comply with our portability-affordability constraint.

The configuration of the New Caledonia region which is currently used for daily forecasts is built on a 1/6° resolution grid encompassing the whole southwest Pacific. A nested domain for grid refinement (5 km) over the New Caledonia Exclusive Economic Zone has been implemented and tested (Figure 4 and section below "A low-cost observatory for survey and validation"), and should be operational in a future system upgrade.

Forecast_tools

The impact of climate change on the coastal oceans and operational oceanography are two of the highlights of modern oceanography. ROMSTOOLS allows running both inter-annual and real-time simulations. In the latter case, we rely on operational global ocean circulation models for the initial and lateral boundary conditions and operational global atmospheric models (introduced through bulk formulations) for surface forcing. Alternatively, we can use a regional atmospheric model to provide small-scale surface forcing data (cf. section below "The atmospheric system" and Figure 2).

In order to limit the volume of data transferred over the Internet, we use the Open-source Project for a Network Data Access Protocol (OPeNDAP) and extract only the necessary subgrid. This method reduces by a great amount the bandwidth needed for data download. This point is critical for laboratories which do not have access to a high performance Internet connection. Another critical point of the same type is the availability of data. It should be acknowledged that U.S. data bases are far more interesting in this respect than European ones, although the efforts made by Mercator (partly under IRD's demand) to provide its quarter-degree resolution forecast product through an OPeNDAP server is encouraging.

If large scale ocean dynamics are slow in comparison with those of the regional system (this might be incorrect near rapidly changing large scale systems such as western boundary currents), the lateral boundary conditions can be interpolated from a coarse-resolution operational model. The issue of the global model resolution becomes more sensitive when considering the initialization problem. In a regional domain with low intrinsic variability where the circulation is directly forced by surface fluxes (for example in the nearshore zone of coastal upwelling regions; or in the deep sea regions away from fronts and eddies), initialization is of lower importance (a restart file created at a given forecast cycle can be used for the next cycle). In this case, ocean circulation predictability is relying on that of the forcing fields. Those are generated using extracted data from the half-degree, 6-hourly Global Forecast System (GFS), freely available via OPeNDAP by the U.S. National Center for Environmental Prediction (NCEP/NOAA). Higher resolution winds from our regional atmospheric model can also be used, as in the lagoon component of the system (See sections below "The atmospheric system" and "Coupling").

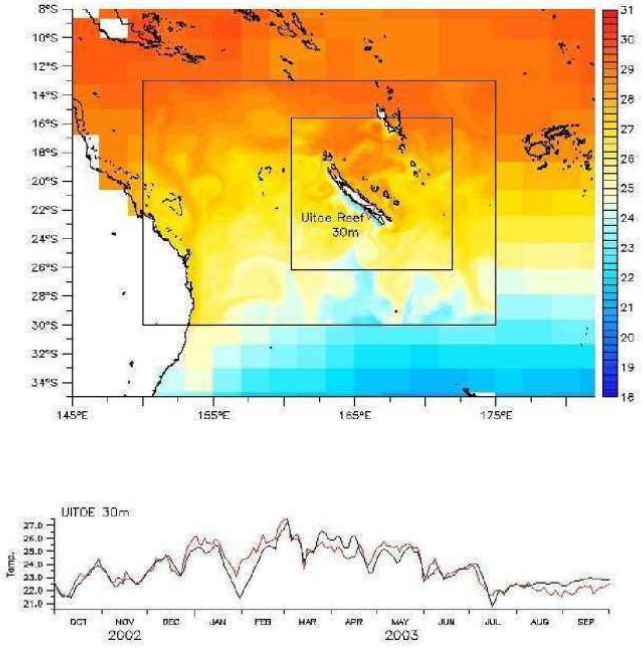


Figure 4

ROMS embedded in Mercator Ocean. Over New Caledonia EEZ, a fine grid at 5km resolution is nested in the parent ROMS domain at 1/6° Top: model SST for January 30, 2003 . Bottom: comparison of model temperature at 30 m with Uitoé station.

The problem becomes more complicated if oceanic intrinsic variability is dominant. In this case, mesoscale eddies, with dynamics evolving at a time-scale of the order of the month, are generated on a regular basis and have low predictability. The model may well provide a very statistically reliable image of the mesoscale dynamics (Marchesiello et al., 2003), but eddies at

any given time are usually not correlated with real events. An initialization method is therefore needed to adjust the statistically reliable oceanic variable to the real time. To that end, a now widely used method is data assimilation. For our nested version of ROMS, a three-dimensional variational system called ROMS_DAS (Li et al., 2008) was developed at the Jet Propulsion Laboratory during a collaborative project involving UCLA (some of the authors, now at IRD, participated in this project). The project called AOSNII consisted in evaluating an adaptive ocean sampling design (Chao et al., 2003), where observations and model solutions regularly interact in order to optimize the survey sampling. Despite the good performances shown, ROMS_DAS has not yet been incorporated in the New Caledonia operational experiment. Instead, we have opted for an alternative method, which makes use of Mercator's new quarter-degree resolution forecast product. The global product in this case assimilates for us real-time mesoscale eddies from satellite altimetry and in-situ data. Then, *Newtonian nudging* is used to adjust Mercator data to ROMS dynamics. The nudging is performed during the nowcast step, and a few iterative cycles can be done to guarantee convergence of the resulting nowcast field (for example when first starting up the operational system).

The strategy for a nowcast/forecast cycle is as follows. A first week of simulation (from t_0-7 to t_0, t_0 being the present time) in hindcast mode is run using interpolated data from Mercator for lateral boundary conditions and GFS for surface forcing. A ROMS restart file (from a previous forecast) is used for initialization at time t_0-7, while nudging assimilates the Mercator global ocean data (with a nudging time-scale of 15 days). This provides a nowcast at t_0, which is the initial condition of the forecast run for the following week. A Shell script (make_roms_forecast.sh) manages the whole procedure: download and pre-processing, hindcast and forecast simulations, post-processing and transfer of graphics to the web page, data storage and preparation of the next nowcast/forecast cycle. To improve the system, the build-in 2-way nesting procedure can be used for local refinement. In this case, the results may be improved in the future by assimilation of small-scale data, as in the AOSNII experiment (Chao et al., 2003).

A low-cost observatory for survey and validation

The oceanographers of the IRD–Nouméa center have developed since 1958 a coastal station network in the southwest tropical Pacific Ocean. Part of this network has been managed in collaboration with the ZoNéCo program since 1992. Originally based on bucket samples, most of the stations are now equipped with automatic ONSET instruments (StowAway TidbiT from HoBo, accuracy +/- 0.2°C) providing temperature time series in the upper 10m depth (the station at Uitoe also gives subsurface temperature and surface salinity data from Seabird SBE16 sensors). The station at the Amédée lighthouse has a real-time connection and we have recently added an automatic process to provide our web site with daily real-time temperature. The in-situ data is completed by real-time satellite images (also automatically deposited on our web site) as well as oceanographic cruises and merchant ship data. These observations can be used for real-time survey, oceanic climate studies, or model validation.

Some validation of the operational system has been performed by Vega (2007) as part of the PREVIMER project. Here, we reproduce a short model-data comparison intended to illustrate the potential usage of our coastal network to assess the benefit of the regional system in resolving sub-mesoscale processes. Of particular interest is the southwest part of New Caledonia barrier reef (station Uitoe), characterised by regular coastal upwelling events. These are produced by trade winds flowing northward along the coast. The Mercator product is unable to capture the upwelling process which requires resolution of 1-10km. Figure 4 presents a snapshot of model SST as well as a 2-year time-series at Uitoé station for both model and station data; the following table (Table 1) also presents some statistics of intra-seasonal variability at various depths:

Onset	Depth	Time serie	Location	ROMS rms	Obs. rms	Bias (°C)	R	Number of Obs
Uitoe 1	10 m	Sep99-Mar-06	21.8°S, 166.1°E	0.65	0.55	-0.72	0.69	633
Uitoe 2	35 m	Jul01-Mar-06	21.8°S, 166.1°E	0.30	0.22	0.26	0.47	816
Uitoe 3	62 m	Jul01-Mar-06	21.8°S, 166.1°E	0.18	0.29	0.25	0.46	669

Table 1

Statistics of intra-seasonal temperature variability at Uitoe station located on the New Caledonia barrier reef. Statistical parameters are: root mean square (rms), bias and correlation (R)

The match between modelled and observed temperatures (in temporal and spatial structure) demonstrate the skills of the regional model in reproducing correctly all the upwelling events. It confirms that coastal upwelling is a predictable process as long as the wind forcing is accurate and the background oceanic conditions are realistic.

The atmospheric system

Similarly to ROMS, the Weather Research & Forecasting (WRF) model (figure 2) is a next-generation regional atmospheric model developed mainly at the National Center of Atmospheric Research (NCAR). WRF is destined to replace the widely used community model MM5 with its advanced numerical methods and parameterizations designed for mesoscale numerical prediction and research. The model comes with WPS (WRF Pre-Processing System) which is a FORTRAN equivalent of ROMSTOOLS able to robustly pre-process global operational model data. In addition, we have developed in a collaborative work between IRD and INRIA a shell-script (make_wrf_forecast.sh) providing an interface between the user and WPS. This simple, yet robust, interface manages data downloads and files system, defines pathways and environment variables, generates all various *namelist* files for WPS executables (geogrid.exe, ungrib.exe, metgrid.exe), run the WPS executables, generates namelist files for WRF and run WRF. Finally, another Shell script file calls for matlab routines to analyze and plot the results, then transfer the graphics to the operational web page.

WRF configuration for the New Caledonia region uses 3-level, 2-way nested domains spanning the area of New-Caledonia Exclusive Economic Zone (20km resolution) with refinement of the horizontal resolution over the Southern Province (6km), and Nouméa area (2km resolution). Terrestrial data are provided by the US Geological Survey, apart from the 50m topography of New Caledonia, which is provided by New Caledonia's Geographical Information System service (SGVL). The simulation starts one day before present using initial and boundary data interpolated from the half-degree, 6-hourly Global Data Assimilation System (GDAS). Next, a 4 day forecast is conducted using boundary data from the half-degree, 3-hourly Global Forecast System (GFS). Note that the time lag (forward in New Caledonia) with the U.S. time is to be taken into account.

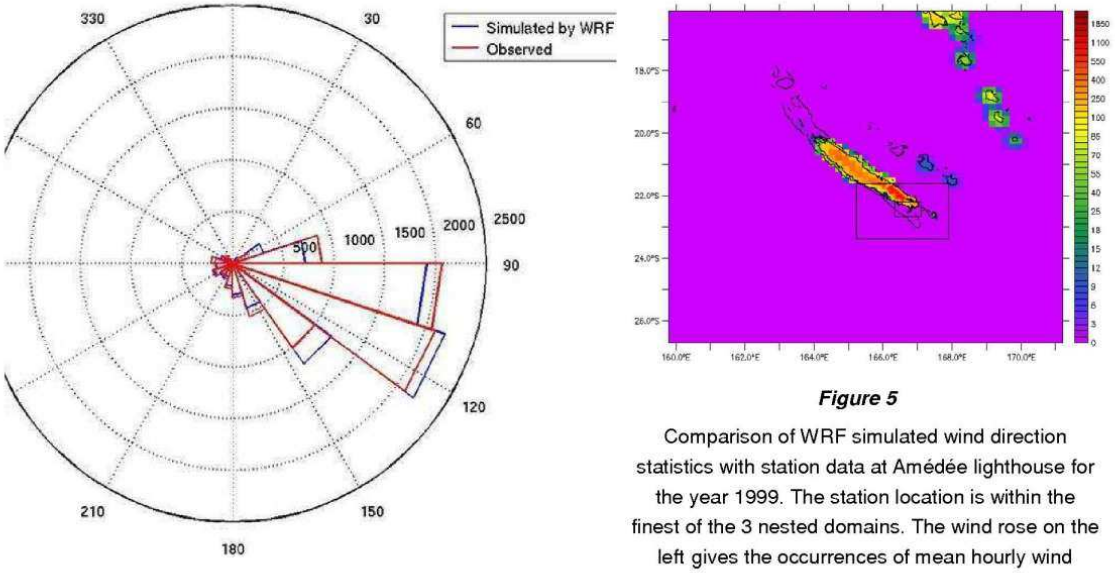


Figure 5

Comparison of WRF simulated wind direction statistics with station data at Amédée lighthouse for the year 1999. The station location is within the finest of the 3 nested domains. The wind rose on the left gives the occurrences of mean hourly wind direction in 18 quadrants of directions.

The validation experiment performed on yearly simulations using the same 3-level nested downscaling system has shown very good results. Figure 5 illustrates the excellent statistical comparisons which partly result from the configuration setup. The model domain is small enough that most of the synoptic variability is generated by the global model while the regional model is used optimally for its capability to produce fine scales from local topographic control and local scale thermal winds.

Coupling

While gaining control of both oceanic and atmospheric models, the idea of coupling them has naturally emerged in the scope of our research and operational projects. Rather than using generic couplers (OASIS for example), which can be difficult to implement and therefore goes against our principle of portability, a specific coupler has been devised for ROMS-WRF (Lemarié et al., 2008). The application of the coupling method to our nowcast/forecast system is in principle straightforward and will be implemented in the future.

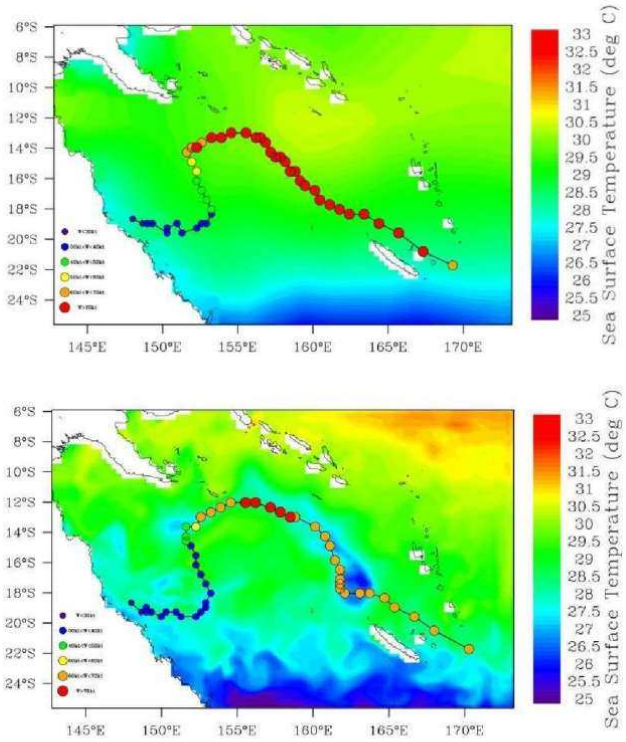


Figure 6

WRF Forced simulation (top) and ROMS-WRF coupled simulation (bottom) of Cyclone Erica moving over New Caledonia in March 2003 (Lemarié et al., 2008). Cold waters are produced on the cyclone track due to strong surface mixing induced by the hurricane-force winds; the cold water induces a negative feedback on the tropical cyclone which is weaker and more realistic as a results (orange rather than red color). Note that the real cyclone path is similar to the simulated one except that it crosses New Caledonia about 200km further south.

Actual 2-way coupling algorithms for realistic high-resolution regional applications are not numerous; examples can be found in Bao et al. (2000). The approach used for climate applications with low coupling frequency are not appropriate for generally fast modes regional applications, and may generate large inaccuracies in the numerical solutions. On the other hand, high frequency coupling (at the ocean time step) may also have its drawbacks as estimated turbulent fluxes can be uncertain on time scales lower than 10 minutes. The choice is either to implement more detailed physical processes relevant to high temporal frequency coupling, such as spray contributions to heat fluxes and the wave boundary layer (Bao et al., 2000), and/or to improve asynchronous methods. In the latter case, it may be useful to impose various coupling properties, in particular the convergence of oceanic and atmospheric fluxes at the interface as well as flux conservation between the two systems. In our coupling algorithm, an iterative method based on domain decomposition (Global-in-time non-overlapping Schwarz methods) was implemented to obtain convergence between fluxes. A FORTRAN package manages the calls to oceanic and atmospheric models and their coupling. It has been applied successfully to the coupled simulation of real tropical cyclones showing the role of oceanic feedback in limiting the potential growth of tropical cyclones (Figure 6; Lemarié et al., 2008).

The lagoon

Lagoons are very specific bodies of sea water, often largely isolated from oceanic waters by a barrier reef, with only a limited number of passes to make the connection. The Lagoon of New Caledonia is the largest closed Lagoon in the world with large biological diversity and degrees of endemism. As such it is a valuable entity to be understood, surveyed and protected. The Lagoon is shallow and its circulation is dominated by barotropic tides and direct wind influences. Therefore, its modeling requires accurate tidal and wind forcing. For tides, freely available data sets such as those from the global model of ocean tides TPXO (version 6 or 7), assimilating satellite altimetry, have proved to be very valuable in many regions, including New Caledonia. Accurate wind forcing at the lagoon scale can only be produced by regional numerical models of which WRF is a good example.

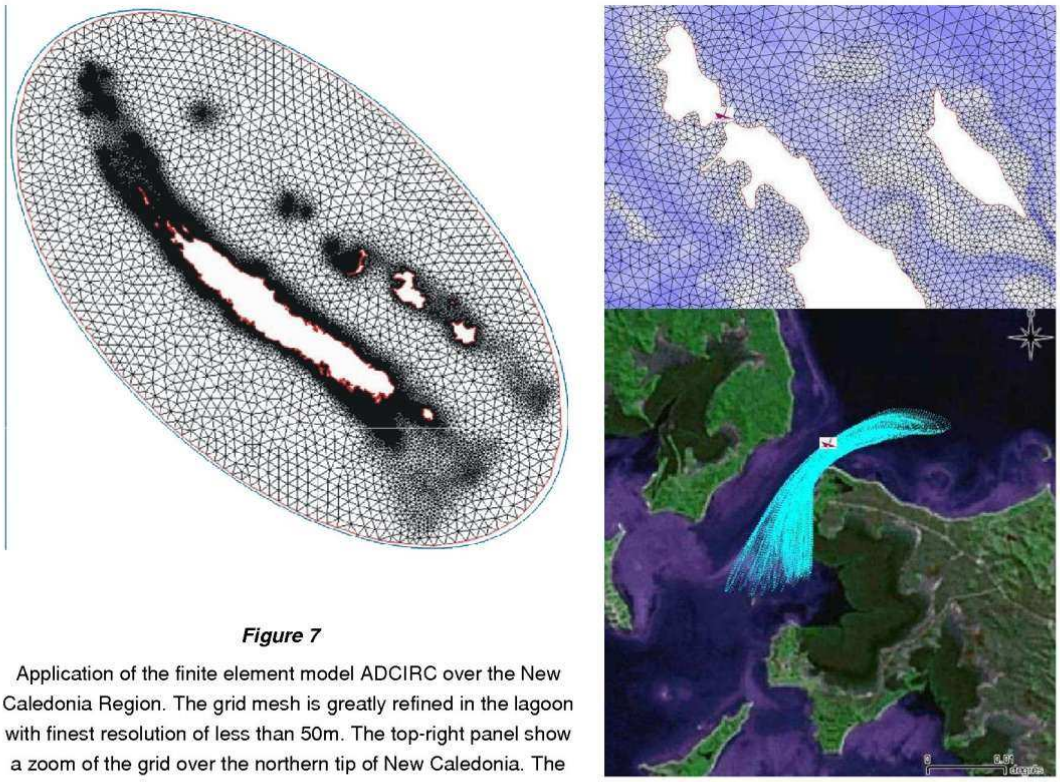


Figure 7

Application of the finite element model ADCIRC over the New Caledonia Region. The grid mesh is greatly refined in the lagoon with finest resolution of less than 50m. The top-right panel show a zoom of the grid over the northern tip of New Caledonia. The lower panel shows a simulation of oil spill during 48 hours after a virtual wreck at Boat Pass.

Three different oceanic models have been used for the New Caledonia lagoon (Figure 2): ROMS, MARS3D and ADCIRC. MARS3D, developed by IFREMER, is historically the first code applied to the lagoon. Its local IRD version benefits from a long experience, where validation against in-situ data (collected during observation programs) permitted fine parameter tuning. MARS3D is in essence similar to ROMS but with less emphasis on numerical aspects as it is traditionally used in shorter term, engineering-type applications. For an operational version of the lagoon model, we used a modified version of ROMS forecast toolbox which provides tidal forcing (from TPXO or local data), surface forcing (from WRF), as well as sub-tidal oceanic forcing (from ROMS oceanic application). This MARS3D operational version is only implemented to the southwest part of the lagoon where most human activities are concentrated.

With the announced spreading of human activity over the whole highland, a new priority is given for modeling and forecasting the whole lagoon. It appears that finite-difference models become limited to that end and we have turned to community finite element models such as the ADvanced CIRCulation model (ADCIRC) (figure 7). If those have not yet proven totally reliable to handle three-dimensional oceanic dynamics (large efforts are currently devoted to that), their successful application to shallow water dynamics have long been recognized. The unstructured grid mesh is a classical feature of these models and is particularly interesting to locally refine the resolution with considerable flexibility. Using TPXO6 tidal data and local high resolution topographic data, we have been able to reproduce tidal elevation and currents with an accuracy and resolution that have never

been achieved before (less than 50m locally). The system is further refined by employing our WRF simulations for small-scale surface forcing.

Technology transfer in Peru with local specifications

Several IRD teams in South America have already benefited from the developments of the above described system. In particular in Peru, several steps towards the implementation of a regional marine forecast system have been recently completed. First, a PC cluster platform based on the Nouméa prototype has been funded by IRD's computing support department (DSI: Direction des Resources Informatique) and implemented at IMARPE (Instituto del Mar del Peru: the Peruvian

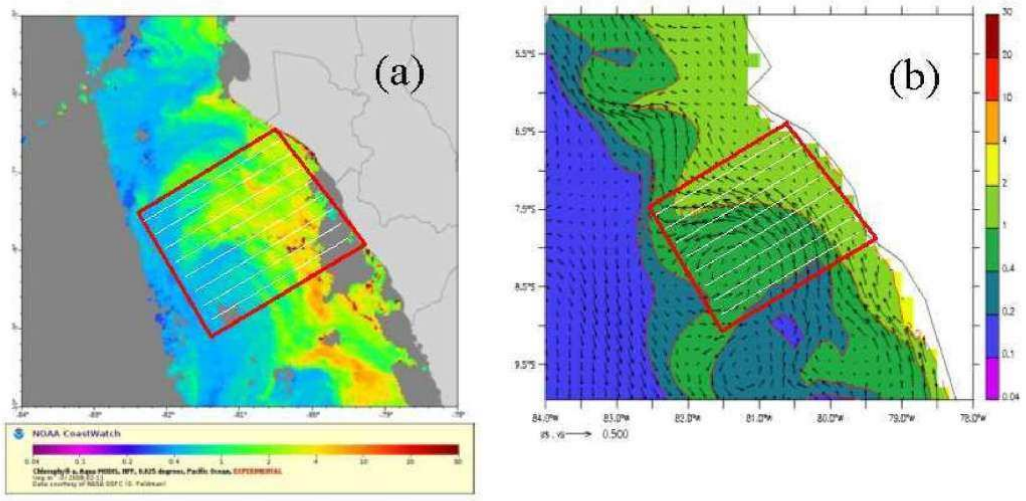


Figure 8

Surface chlorophyll-a (mg Chl/m3) off Peru. (a) Satellite observations in February 2008 (courtesy of F. Chavez, MBARI). (b) ROMS-PISCES (1/6°) chlorophyll and currents during a similar upwelling event. The model is initialized and forced by the Global 1/4° monovariate Mercator product. The white lines show the "filamentos" cruise track. The cruise data and operational modeling system will be used jointly during the MESUP project to further refine the biogeochemical module parameters and improve prediction capabilities.

marine research institute). Those improved computing capabilities are needed to carry on-going modeling projects, which include the PCCC project (Peru-Chile Climate Change, funded by the French ANR) and research activities dedicated to biogeochemical modeling (using the module PISCES which is implemented in ROMS). Those modeling efforts are made in re-analysis (hindcast) rather than real-time and forecast mode, but they rely on the same forcing strategy, in particular the Mercator products for initialization and lateral forcing at the open-boundaries of the regional oceanic model configuration. Recently, in conjunction with an observational field operation off Chimbote (9°S) dedicated to the study of a cold upwelling filament structure ('Filamentos' cruise), a modeling project was proposed and funded by Mercator to simulate the filament and its associated biological response (project MESUP) (Figure 8).

Other projects at IMARPE are focused on the equatorial Kelvin wave impact on the regional circulation off Peru. They make use of equatorial wave models (that are suitable for developing countries since they permit generating their own large scale products without the requirement of a complex ocean general circulation model) and Mercator outputs that can be used to derive an estimation of the equatorial Kelvin wave contribution (see Dewitte et al. 2007 as well as Illig et al. 2007). Finally, a near real-time application of the modeling tools will be experienced at the occasion of a regional cruise, which is planned for October 2008 off the coast of Central Peru (in the frame of the VOCALS international program). Clearly, the Peruvian IMARPE-IRD projects tend to confirm that the development of 'handy' modeling platforms, in parallel with affordable large scale assimilation products, are the warrant for a sustainable evolution of the collaborations with developing countries.

Assessment of system usefulness and conclusion

The assessment of system usefulness is a key ingredient in motivating our interest in operational coastal and regional oceanography. Since our prototype system for New Caledonia has been known for 2 years. we are now able to synthesize local information and user feedback. Potential users in New Caledonia are variably interested in the different component of the

system (ocean, atmosphere, and lagoon). The high resolution wind forecast on its own appears to satisfy the largest demands, especially as no other high resolution product is available (Météo-France does not have any refinement strategy in the region and relies on a good observational network and global forecast from Toulouse). The wind forecast has first been used by our scientist colleagues, when field experiment conditions in the lagoon are strongly affected by small-scale winds; it has then been known also by professionals proposing sailing or diving services, or by private persons practicing leisure activities in the lagoon (including us), and a demand seems to emerge from wind power companies which relies importantly on local wind prediction. The interest of oceanic survey and forecast outside the lagoon appears only obvious for local small fishing companies looking for pelagic fish whose presence is highly dependent on oceanic environmental conditions (tuna in particular). This part of the system has also been of use to oceanographic cruises on occasions. The potential usefulness of lagoon forecast is expected to be major, provided that additional information is given. For example, the New Caledonia Anti-Pollution Comity (which assembles petrol and mining companies together with governmental agencies to optimize emergency actions against pollution) would be more interested in our product if a drift model for oil spill or chemical pollution is coupled to the lagoon circulation model.

More generally, there are a number of application modules that can improve the usefulness of operational systems: *oil drift* modules integrating wind and current effects; *biogeochemical* and *sediment* transport modules for assessing the risks of biochemical pollution; biological modules to predict larval connectivity for optimal monitoring of marine reserves (see for example the IBM model implemented during the PREVIMER project by Lett et al., 2008); other types of biological modules to predict fish population density as a function of environmental variables and fishing pressure (see on-going implementation of the SEAPODYM tuna model for New Caledonia); modules assessing the risks of *extreme events* (cyclones) or even *emerging infectious diseases* (malaria and dengue are a major health concern in tropical countries; an IRD diagnostic module exists that has been tested for dengue risks assessment in New Caledonia, using WRF predictions).

Finally, the relevance of an operational marine system is dependent on the type of environmental events. Our present system for example is not covering oceanic processes relevant to high-frequency surface conditions such as swell and tsunamis. Swell prediction should eventually be incorporated into the system, as swell can greatly affect the flushing of the lagoon; tsunami prediction relies greatly on international networks of observation and alert, as well as local expensive moored stations. In conclusion, if progress can be made to meet some hypothetic usefulness standards, the present first approach towards building affordable marine forecast systems has reach its main objectives and leaves us with an encouraging prospect.

References

Bao et al., 2000 J. Bao, J.M. Wilczak, J. Choi and L.H. Kantha, 2000: Numerical simulations of air–sea interaction under height wind conditions using a coupled model: a study of hurricane development, *Monthly Weather Review* 128, pp. 2190–2210.

Blaas, M., C. Dong, P. Marchesiello, J.C. McWilliams, K.D. Stolzenbach, 2007: Sediment-transport modeling on Southern Californian shelves: A ROMS case study. Continental Shelf Research, 27, 832-853.

Blanke, B., C. Roy, P. Penven, S. Speich, J. McWilliams, and G. Nelson, 2002: Linking wind and interannual upwelling variability in a regional model of the southern Benguela, Geophys. Res. Lett., 29(24), 2188, doi:10.1029/2002GL015718.

Capet X.J. ,P. Marchesiello, and J.C. McWilliams, 2004: Upwelling response to coastal wind profiles. Geophysical Research Letters 13, L13311, doi: 10.1029/2004GL020123.

Chao, Y, Z. Li, J. Choi, P. Li, J. McWilliams, P. Marchesiello, and X. Capet, 2003: A Real-Time Operational Forecasting System for the Central California Coastal Ocean. 2003 Eastern Pacific Ocean Conference (EPOC), September 2003.

Dewitte B., M. Ramos, O. Pizarro and G. Garric, 2007, Connexion between the equatorial Kelvin wave and the extra-tropical Rossby wave in the South-Eastern Pacific in the Mercator POG05B Simulation: a case study for the 1997/98 El Niño, Mercator Ocean Newsletter n°26, July 2007, http://www.mercat or-ocean.fr/documents/lettre/lettre_26_en.pdf

Karakas, G., N. Nowald, M. Blaas, P. Marchesiello, S. Frickenhaus, and R. Schlitzer, 2006: High-resolution modeling of sediment erosion and particle transport across the northwest African shelf, J. Geophys. Res., 111, C06025, doi:10.1029/2005JC003296.

Gruber, N., H. Frenzel, S.C. Doney, P. Marchesiello, J. C. McWilliams, J. R. Moisan, J. Oram, G.-K., Plattner, and K.D. Stolzenbach, 2006: Simulation of phytoplankton ecosystem dynamics in the California Current System. Deep-Sea Research I, 53, 1483-1516.

Illig S., B. Dewitte, C. Périgaud and C. Derval , 2007, Equatorial wave intra-seasonal variability in the Indian and Pacific Oceans in the Mercator POG05B Simulation, Mercator Ocean Newsletter n°26, July 2007, http://www.mercator-ocean.fr/documents/lettre/lettre_26_en.pdf

ANNEXE B Présentation abrégée du switch InfiniBand Flextronics et des composants IBA : carte InfiniHost et câbles

(source : http://ibswitches.com/Docs)

IBSwitches.com
The Infiniband Price / Performance Leader

Contact Us for a quote
info@ibswitches.com

Home | Products | Applications | Resource Center | Support | Contact Us

Featured Product

Flextronics 24 Port Infiniband DDR Switch

Description

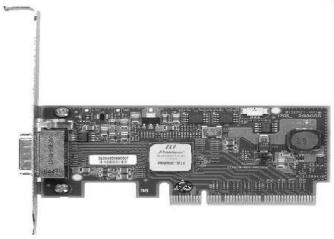
The 24 Port Infiniband DDR Cluster Kit includes 24 Mellanox branded Single Port Infiniband DDR HCA's, a Flextronics 24 Port DDR Switch, and 24 Mellanox passive copper Infiniband cables. This cluster kit in a box provides all the necessary components required to interconnect 24 servers in to a high performing Infiniband DDR cluster. All components are genuinely branded Mellanox and Flextronics equipment carrying full 1 year warranties.

Key Features

- *Modular 24 Port 4X DDR*
- *24 Single Port Mellanox DDR HCA's*
- *24 Mellanox Passive Copper Cables*

Flextronics 24 Port DDR Switch

This Flextronics 24 port 4X non-blocking DDR Infiniband switch is based on Mellanox Infiniscale III silicon technolgy. Whether you are interested in building small department clusters or want to build a Top 500 cluster, Flextronics switches are the most cost effective and highest performing Infiniband switches in the market.



Single Port Mellanox DDR HCA's

Included in the kit are 24 of the Mellanox MHGS18-XTC Single Port DDR Infiniband HCA's. These are PCIe X8 Infiniband HCA's operating at 20Gb/sec. Mellanox is the leading supplier of high performance Infiniband products.

Mellanox Infiniband Cables

Mellanox cables are a cost effective solution for connecting InfiniBand fabrics, extending the benefits of Mellanox's high-performance InfiniBand HCAs throughout the network. These cables meet or exceed the IBTA and IEEE standards, and have been rigorously tested with all Mellanox products.

Custom Configurations Available

All Cluster kits sold are based on genuine Mellanox based silicon technology. Custom configurations are also available.
Request Custom Config Quote

Infiniband 24 Port DDR Cluster Kit

Benefits .....

- Lowest Cost Per Port
- Proven DDR Technology
- Industry Leading Bandwidth
- Based on Flex Switches and Mellanox HCA's
- Ultra low latency

Key Features .....

- Kit includes all HCA's, Cables, and Switch to build 24 Port DDR Infiniband Cluster
- Fully redundant Power and cooling configurations
- 960 Gb/sec bandwidth
- 1 Year warranty with optional extended warranty support

FLEXTRONICS
Authorized Representative

Copyright © 2008, IB Switches.com
All rights Reserved.

ANNEXE C Installation du cluster ToolKit OSCAR : événements postés sur les forums OSCAR

12/01/2005 x86_64 archi : ROCKS & OSCAR partial experience
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04265.html

12/01/2005 x86_64 : Redhat WS (license) + Oscar 4.1b availability
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04266.html

13/01/05 x86_64 & FC 2 : Oscar RPMs rebuild
http://www.mail-archive.com/oscar-devel@lists.sourceforge.net/msg02976.html

20/01/2005 FC 2 i386 : Build Image fail (tftp)
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04329.html

20/01/2005 RE: Continue : FC 2 i386 : Build Image fail (tftp)
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04338.html

20/01/2005 Suite : FC 2 i386 : Client install + systemimager
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04348.html

20/01/2005 Suite : FC 2 i386 : Client install (undefined Disk0)
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04346.html

23/01/2005 Build Image fail : My oscarinstall.log
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04365.html

23/01/2005 SUCCESS : FC 2 i386 : Build Image fail (tftp)
http://www.mail-archive.com/oscar-devel@lists.sourceforge.net/msg03028.html

23/01/2005 ERRATA : [Oscar-users] SUCCESS : FC 2 i386 : Build Image fail (tftp)
http://www.mail-archive.com/oscar-devel@lists.sourceforge.net/msg03029.html

25/01/2005 FC 2 i386 : install node on SIL3114 drive
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04380.html

21/03/2005 lamboot not found
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg04666.html

28/03/2006 RE: 4.2.1b54423 : Post-install : PBS_SERVER and MAUI trouble
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg05389.html

28/03/2006 4.2.1b54423 : PBS_SERVER VS MAUI
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg05394.html

29/03/2006 4.2.1b54423 : MAUI update 3.2.6p14 SUCCESS
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg05408.html

10/02/2008 ydl-5-ppc64 + Oscar on Xserve G5 ?
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg06753.html

11/02/2008 Oscar 5.1 + FC8 on Tyan cluster : short report
http://www.mail-archive.com/oscar-users@lists.sourceforge.net/msg06756.html

ANNEXE D Présentation abrégée du paquetage Mellanox OpenFabrics Enterprise Distribution for Linux (MLNX_OFED)

(source : http://www.mellanox.com/related-docs/prod_software/MLNX_OFED_1_4_PB.pdf)

Mellanox TECHNOLOGIES

SOFTWARE

OFED (OpenFabrics Enterprise Distribution)

High-Performance Server and Storage Connectivity Software for Field-Proven RDMA and Transport Offload Hardware Solutions

Use of clustered commodity servers, in lieu of traditional supercomputers and mainframes, offers tremendous price/performance benefits and unparalleled flexibility in deployment and long-term maintenance. To enable distributed computing transparently, high-performance computing (HPC) applications require the highest bandwidth and lowest possible latency. In enterprise data center (EDC) applications, these requirements are compounded with the need to support a large interoperable ecosystem of networking, virtualization, storage, and other applications and interfaces.

The OFED from OpenFabrics Alliance (www.openfabrics.org) has been hardened through collaborative development and testing by all major InfiniBand vendors. OFED is supported by Mellanox and major InfiniBand vendors to enable OEMs and System Integrators to meet the needs of HPC and EDC applications.

Higher I/O Service Levels for all Data Center Traffic and Applications

For HPC applications, OFED offers Message Passing Interface (MPI) implementation from Ohio State University (OSU) and Open MPI (www.open-mpi.org). Bandwidth results in excess of 3200MB/s and application latency of 1 microsecond have been achieved. They include features to enable scaling to large clusters while improving on memory usage and latency related efficiencies.

For traditional sockets-based applications, OFED includes the field proven implementation of IP-over-IB, enabling IP-based applications to work seamlessly over InfiniBand and perform at levels more than six times higher than Ethernet. It also includes the IBTA (www.InfiniBandTA.org) defined Sockets Direct Protocol (SDP) enabling traditional TCP/IP sockets-based applications to capitalize on the RDMA and transport offload capabilities of InfiniBand. Reliable Datagram Services (RDS) provides an efficient sockets-based interface for Oracle RAC database applications.

To enable traditional SCSI and iSCSI-based storage applications to enjoy similar RDMA performance benefits, OFED includes the SCSI over RDMA Protocol (SRP) initiator and target, and the iSCSI RDMA Protocol (iSER) that interoperate with various target components available in the industry. SRP solutions over InfiniBand have been proven to deliver impressive 910MB/s (random read) and 725MB/s (random write) I/O storage performance with 1MB blocks.

High Availability (HA) Support for Mission Critical Deployments

OFED 1.4 includes high availability support for sockets and storage applications. Specifically, the standard Linux channel bonding module is supported over IPoIB enabling failover across ports on the same adapter, or across adapters. Similarly, standard multipathing and failover over the SRP Initiator is supported for SCSI applications using standard Linux implementations such as device mapper multipath (dm-multipath) driver. Some vendor-specific fail-over/load-balancing driver models are supported as well.

BENEFITS

- Single software stack that operates across all available InfiniBand devices and configurations such as mem-free, QDR/DDR/SDR, PCI-X, and PCI Express modes (Gen1 & Gen2)
- Support for HPC applications for scientific research, oil and gas exploration, car crash tests, bench marking etc. E.g., Fluent, LS-DYNA
- Support for EDC applications such as Oracle 11g/10g RAC, IBM DB2, SAP, and financial applications
- VPI software stack allows Mellanox ConnectX devices to run both InfiniBand and 10GigE traffic simultaneously on two ports
- Support for high-performance block storage applications utilizing RDMA benefits
- Support for end-to-end QoS using centralized QoS manager for delivering no compromise unified I/O services
- Support for cluster management tools

SOCKETS LAYER

- SDP and IP-over-IB component enable TCP/IP and sockets-based applications to interface seamlessly to and benefit from InfiniBand transport

ACCESS LAYER

- Supports the OpenFabrics defined Verbs API at the user and kernel levels. User level verbs allow protocols MPI and other applications to interface to Mellanox InfiniBand hardware. Kernel levels verbs allow protocols like SDP, SRP and IP-over-IB (with LSO and LRO support) to interface to Mellanox InfiniBand hardware

SCSI MID LAYER

- The SCSI Mid Layer interface enables SCSI based block storage and management applications to interface with the SRP Initiator component and the Mellanox InfiniBand hardware

Ready for Unified I/O Deployments with Granular Quality of Service (QoS)

OFED QoS features enable data center managers to define and deploy QoS policies end-to-end by traffic type using a centralized QoS manager. All network nodes, including servers, switches and storage end-points, are part of coherent QoS policies, delivering true QoS in a reliable, lossless medium. Centralized management reduces management effort and enables faster definition and deployment of QoS policies that deliver immediate results. Based on IBTA defined QoS extensions, QoS features supported with OFED 1.4 include granular traffic classification, traffic marking, and scalable and easy-to-deploy policing and shaping mechanisms that can be easily implemented through customization of sample QoS configuration files used by the QoS Manager (part of Open Subnet Manager)

OFED is Interoperable Across Multi-Vendor Solutions

As a founding member of OpenFabrics, previously known as OpenIB, Mellanox is driving interoperability of the OpenFabrics software across different vendor solutions. OFED, running on servers, interoperates with InfiniBand switches, gateways and storage targets and value added software provided by vendors in such systems. Today, OFED is distributed as part of major Linux OS and virtualization software distributions. OFED supports all popular CPU platforms.

Enabling a Large Software Ecosystem

Through its collaboration with the OpenFabrics community, support for multiple industry-standard interfaces that enable off-the-shelf applications to work with InfiniBand, and support for popular OS and CPU platforms, Mellanox has created a large, interoperable software ecosystem that benefits the HPC and EDC markets.

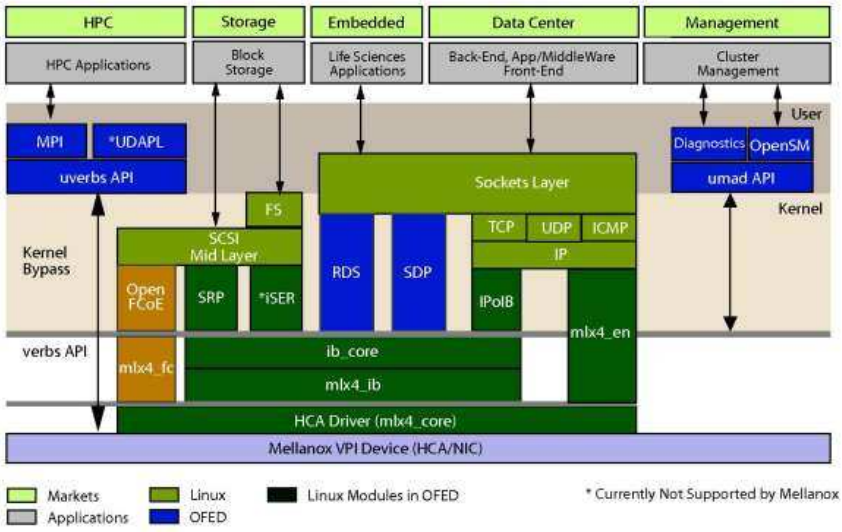
COMPONENTS

- Driver & Access Layers
- VPI (Virtual Protocol Interconnect)
- SRP Target
- OSU MVAPICH and Open MPI
- IP-over-IB
- IP-over-IB (with LSO & LRO support)
- SDP
- SRP Initiator
- iSER Initiator*
- uDAPL*
- RDS
- Subnet Manager (OpenSM)
- Installation, Administration and Diagnostics Tools
- Performance test suites

* Currently not supported by Mellanox

DEVICE SUPPORT

- InfiniHost® Adapter Silicon
- InifiniHost® III Ex /Lx Adapter Silicon
- ConnectX® Adapter Silicon
- Memory & Memory-free Adapter Cards
- InfiniScale® Switch Silicon
- InifiniScale® III Switch Silicon
- InfiniScale IV® Switch Silicon
- MTS3600 InfiniBand Switch



Note: SRP Target is included in the OFED package but is not shown in this diagram

Platform Supplier Products and Support	
RedHat	EL4 up4,up5, up6, up7 EL5 and up1, Fedora C6
Novell	SLES10, SLES10 SP1 and up1, SLES10 SP2, OpenSuSE 10.3
Intel	x86, IA64, EM64T, EM64T x86_64, PCI-X and PCIe platforms
AMD	AMD64 and AMD64-Ex Opteron, PCI-X and PCIe platforms
IBM	PPC64

Mellanox TECHNOLOGIES
350 Oakmead Pkwy, Suite 100, Sunnyvale, CA 94085
Tel: 408-970-3400 • Fax: 408-970-3403
www.mellanox.com

OPENFABRICS ALLIANCE

© Copyright 2009. Mellanox Technologies. All rights reserved.
Mellanox, ConnectX, InfiniBlast, InfiniBridge, InfiniHost, InfiniRISC, InfiniScale, and InfiniPCI are registered trademarks of Mellanox Technologies, Ltd. Virtual Protocol Interconnect is a trademark of Mellanox Technologies, Ltd. All other trademarks are property of their respective owners.

ANNEXE E Fichier de configuration ‘’ofed.conf’’ définissant les modules et drivers IBA à installer dans le processus d’installation MLNX_OFED

Ofed_editr.conf	Ofed_nodes.conf
build32=0 prefix=/usr kernel-ib=y core=y mthca=y nes=y ipath=y ipoib=y sdp=y srp=y srpt=y rds=y kernel-ib-devel=y libibverbs=y libibverbs-devel=y libibverbs-devel-static=y libibverbs-utils=y libibverbs-debuginfo=y libmthca=y libmthca-devel-static=y libmthca-debuginfo=y libmlx4=n libmlx4-devel-static=n libmlx4-debuginfo=n libcxgb3=n libcxgb3-devel=n libcxgb3-debuginfo=n libnes=y libnes-devel-static=y libnes-debuginfo=y libipathverbs=y libipathverbs-devel=y libipathverbs-debuginfo=y libibcm=y libibcm-devel=y libibcm-debuginfo=y libibcommon=y libibcommon-devel=y libibcommon-static=y libibcommon-debuginfo=y libibumad=y libibumad-devel=y libibumad-static=y libibumad-debuginfo=y libibmad=y libibmad-devel=y libibmad-static=y libibmad-debuginfo=y ibsim=y ibsim-debuginfo=y librdmacm=y librdmacm-utils=y librdmacm-devel=y librdmacm-debuginfo=y libsdp=y libsdp-devel=y	prefix=/usr kernel-ib=y kernel-ib-devel=y libibverbs=y libibverbs-devel=y libibverbs-devel-static=y libibverbs-utils=y libmthca=y libmthca-devel-static=y libmlx4=n libmlx4-devel-static=n libcxgb3=n libcxgb3-devel=n libnes=y libnes-devel-static=y libipathverbs=n libipathverbs-devel=n libibcm=y libibcm-devel=y libibcommon=y libibcommon-devel=y libibcommon-static=y libibumad=y libibumad-devel=y libibumad-static=y libibmad=y libibmad-devel=y libibmad-static=y librdmacm=y librdmacm-utils=y librdmacm-devel=y libsdp=y libsdp-devel=y opensm=n opensm-libs=n opensm-devel=n opensm-static=n perftest=y mstflint=y tvflash=y sdpnetstat=y srptools=y rds-tools=y ibutils=y infiniband-diags=y ofed-docs=n ofed-scripts=y mpi-selector=n mvapich_gcc=n mvapich2_gcc=n openmpi_gcc=n mpitests_mvapich_gcc=n mpitests_mvapich2_gcc=n mpitests_openmpi_gcc=n core=y mthca=y

libsdp-debuginfo=y
opensm=y
opensm-libs=y
opensm-devel=y
opensm-debuginfo=y
opensm-static=y
dapl-v1=y
dapl-v1-devel=y
dapl-v2=y
dapl-devel=y
dapl-devel-static=y
dapl-utils=y
dapl-debuginfo=y
perftest=y
mstflint=y
tvflash=y
sdpnetstat=y
srptools=y
rds-tools=y
ibutils=y
infiniband-diags=y
qperf=y
qperf-debuginfo=y
ofed-docs=y
ofed-scripts=y
mpi-selector=y
mvapich_gcc=y
mvapich_pgi=y
mvapich_intel=y
mvapich2_gcc=y
mvapich2_pgi=y
mvapich2_intel=y
openmpi_gcc=y
openmpi_pgi=y
openmpi_intel=y
mpitests_mvapich_gcc=y
mpitests_mvapich_pgi=y
mpitests_mvapich_intel=y
mpitests_mvapich2_gcc=y
mpitests_mvapich2_pgi=y
mpitests_mvapich2_intel=y
mpitests_openmpi_gcc=y
mpitests_openmpi_pgi=y
mpitests_openmpi_intel=y
mvapich2_conf_impl=ofa
mvapich2_conf_romio=1
mvapich2_conf_shared_libs=1
mvapich2_conf_ckpt=0
mvapich2_conf_vcluster=small
mvapich2_conf_dapl_provider=ib0

mlx4=n
cxgb3=n
nes=y
ipath=n
ipoib=y
sdp=y
srp=y
rds=y
qlgc_vnic=y
iser=y
mvapich2_conf_impl=ofa
mvapich2_conf_romio=1
mvapich2_conf_shared_libs=1
mvapich2_conf_multithread=0
mvapich2_conf_ckpt=0
mvapich2_conf_vcluster=small
mvapich2_conf_dapl_provider=ib0